AN ARGUS SPECIALIST PUBLICATION

April 1986

£1.00

Your COMMODORE

YOUR BEST INDEPENDENT COMMODORE MAGAZINE

Now Incorporating YOUR 64

Make a trunk call on your Commodore with Telcom 64



Databos— Our first C128 program

Gremlins in your C-16/Plus 4

APRIL 1986

VOLUME 2
NUMBER 7

Editor:
Stuart Cooke

Assistant Editor:
Marie Curry

Advertisement Manager:
Mike Segrue

Advertisement Copy Control:
Laura Champion

Group Editor:
Dave Bradshaw

Group Managing Editor:
Wendy Palmer

Managing Director:
Peter Welham

Origination:
Ebony Typesetting

Design:
Argus Design

Editorial & Advertisement Office
No 1 Golden Square,
London W1R 3AB.
Telephone: 01-437 0626
Telex: 8811896

Your Commodore is a monthly magazine appearing on the first Friday of each month.

Distribution by: Argus Press Sales & Distribution Ltd. 12-18 Paul Street, London EC2A 4JS. Printed by: Alabaster Passmore & Sons Ltd, Tovil, Maidstone, Kent.

Subscription rates upon application to Your Commodore Subscriptions Department, Infonet Ltd, Times House, 179 The Marlowes, Hemel Hempstead, Herts. HP1 1BB.

CONTENTS

FEATURES

SERIES

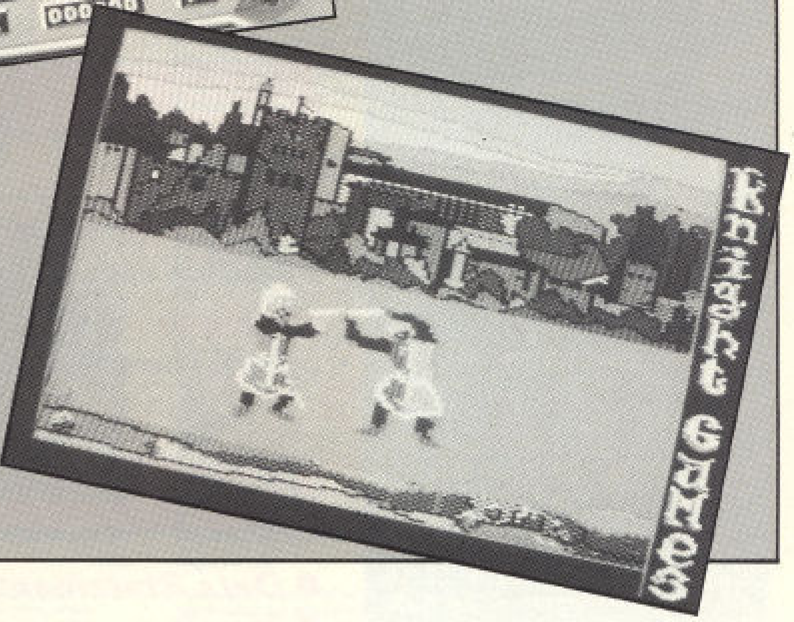
REGULARS

GAMES AND UTILITIES



DATA STATEMENTS





Soft in the Head

GAMES SOFTWARE HAS ALWAYS MANaged to cover the most weird and wonderful themes and the latest new releases are no exception.

There are two new film spin-offs just released on to the market and the first, from Electric Dreams Software, is based on Steven Spielberg's recently highly successful fantasy movie – Back to the Future. Rod Cousens, Electric Dreams' MD, makes great claims for the new game. "It is a strong game in its own right, designed to complement the film – we haven't just taken the name of the film." Perhaps the most innovative aspect of the new game is that it features teenage romance and even an on-screen kiss!

Back to the Future closely follows the storyline of the movie in which the hero, Marty, arrives in the 1950s and finds himself having to engineer a romantic meeting between his parents, Lorraine and George. Certainly not your average zap-em-up! To enter this coffee bar world of teenage dreams will cost you a mere £9.95 and you only need one qualification – a C64.

Still on the subject of films, it's classic horror this time. Remember The Phantom of the Opera, that ghastly hunchback who roamed the Paris sewers at the turn of the century? He's now been resurrected for the fourth time only the screen's a bit smaller this time round. There have been three film versions of this chilling tale, in 1931, 1943 and 1962. 1986 sees The Phantom's big revival with not only a computer game, but also an Andrew Lloyd Webber stage musical.

This one's also a love story but in an entirely different vein. A horribly disfigured musician falls for a beautiful opera singer. Not exactly an ideal match because she's not in the least bit interested. Not deterred by her revulsion the Phantom takes the bull by the horns and abducts her. From there it's a classic damsel in distress style rescue plot. Popular opera themes bring a touch of class and author John Ransley calls it "Hunchback in evening dress". If you've got a C64 you can find out for yourself.

Melbourne House recently announced the release of the epic sequel to The Hobbit. This of course is the Lord of The Rings Part I. Because of the enormous size of the program, the game has had to be released on two cassettes and the package also includes a copy of the paperback, The Fellowship to the Ring, plus an extensive instruction booklet.

Melbourne House claims many outstanding features for the game including its ability to understand Inglish – let's hope its spelling isn't always that bad! With Inglish, apparently, you can now say almost anything you want because the computer can recognise intelligent sentences up to 128 characters long, pretty impressive. There is a large degree of interaction with the other characters, you can answer and tell them what to do. There are over 200 locations, a vocabulary of 800 words and even a multi-player option. It's available on the C64 on both cassette and disk, £15.95 and £19.95 respectively. If you're really into the Tolkien cult then you might also be interested in the Lord of the Rings hologram offer. Each hologram features the image of a Nazgul, or ring wraith – commonly known as a Black Rider. A small hologram – 5" × 4" – costs £29.95 including p&p and there is a limited edition of a large version – 100 copies only – for a staggering £149.95.

English Software has just released two new titles which are as far apart in concept as is possible. Knight Games is a medieval tournament type game where you compete in seven ancient and very violent events: Sword fighting, duelling axemen, archery, quaterstaff bashing, crossbow shooting, ball and chain flailing and pikestaff contests. Definitely Daley Thompson with a difference.

The other English Software release is Elektraglide, a fantasy racing game featuring tunnels, hair-raising bends, scrolling graphics and a rock music soundtrack. These excursions into the future and the past are for the C64 and both will cost you £8.95 on cassette. Elektraglide is also available on disk at £12.95.

Mastertronic has launched itself into the new year with a flourish. After a very successful 1985, there are several new

releases on the way which Mastertronic hopes will continue its success in the budget software market.

There are three new Mastertronic games for Commodore machines – two are for the C64 and one for the C-16. Kane is another wild west adventure, which seems all of a sudden to have become a popular games genre. Mastertronic says: "It makes High Noon look like a 12 o'clock lunch break."

Mastertronic is also going for this year's oddest game title award with another C64 release, dozily entitled Zzzz! Hopefully it's not a game guaranteed to send you to sleep.

The C-16 game is called Big Mac and that's all we can tell you at the moment because Mastertronic has labelled it top secret. Should be a meaty one though, so look forward to it with relish! All three are in the £1.99 range but they're not all edible.

Hewson Consultants have officially changed their name to plain old Hewson and are sporting a new image for the new year. Spearheading Hewson's new releases for 1986 is Uridium, the sequel to Paradroid, which is reviewed in this issue of Your Commodore. Turn to our Action Replay section to find out what we thought of it.

There's also a very high-tech setting for Beyond's latest title Quake Minus One. A terrorist group called the Robot Liberation Front are trying to sabotage the Titan Power Station which is situated in the depths of the Atlantic Ocean and supplies all the world's industrial nations with energy. There are five computers down there and four of them are in the hands of the RLF. That's what comes of putting all your eggs in one basket. You've got to regain control of the electronic installation from the hands of the metallic malefactors. It's on the Monolith label at £9.95 and is also featured in this month's Action Replay.

Also from Beyond, this time by Denton Design, is Enigma Force, the C64 sequel to Shadowfire. Five members of the Enigma team – all with totally unpronounceable names – are escorting the evil General Zoff to face the consequences of his life of crime. When war break's out, the spaceship crashes and the Enigma Force are faced with the daunting task of killing Zoff, finding the only spaceworthy ship on the planet and escaping. Will it be too much for even their awesome powers? Spend £9.95 and find out.

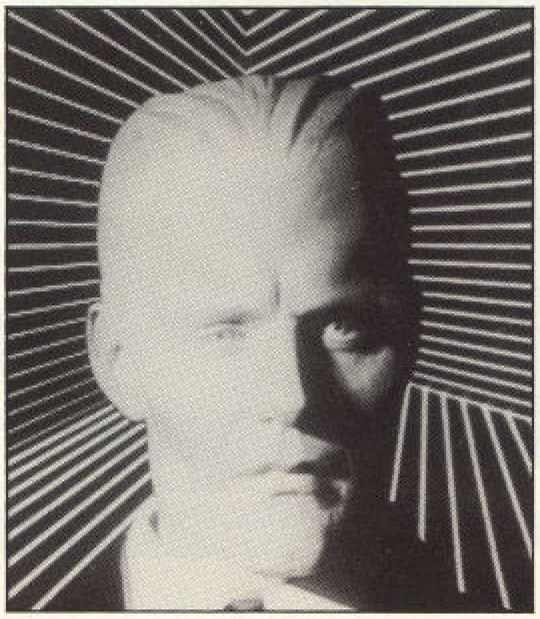
Activision has now released Mindshadow on the C64. It's a graphic adventure which will be a nightmare for amnesiacs. You awake on a deserted beach with no memory and an uncertain future. If you decide not to take a quick holiday by the sea, then you should go in search of your identity. There are logic puzzles for experienced adventurers, 80 screens with detailed graphics and some with animation, an on-screen adventure tutorial for beginners and a cheat book if you get stuck on the hard bits.



Koronis Rift is another in Activision's series of Lucas Film games. It's about a wandering technoscavenger who discovers the legendary Koronis Rift – a weapons testing ground of the ancients, presumably a latterday Bikini Atoll. It's littered with valuable junk which is yours for the taking, provided that you don't get wasted by the ancient guardians – £9.99 on C64 cassette.

Now a couple of quickies from CRL – there are two new C64 games out, Space Doubt and Causes of Chaos, both are on sale now at £8.95. There's also been an official leak from CRL to say that a Nu-Wave label is to be established which will feature alternative software. The release is in March and will be entitled Darkness at Dawn. What's new about it? It's all sound and no graphics. Most odd – definitely different.

And for Max Headroom fans – the game based on the film of the birth of the first computer generated TV presenter will soon be appearing on your computer screen. The game is from Quicksilva and plots the creation and subsequent career of sarcastic Max. It'll be out on the C64.



Touchline

Mindshadow, Koronis Rift: Activision, 5/7 Forelease Rd, Maidenhead, Berks SL6 1RP.
Enigma Force, Quake Minus 1: Beyond, 3rd Floor, Lector Court, 151 Farringdon Rd, London EC1R 3AD. 01 837 2899.
Paradroid: Hewson, Hewson House, 56B Milton Trading Estate, Milton, Abingdon Oxon OX14 4RX. 0235 832939.
Big Mac, Zzzz, Kane: Mastertronic 8-10 Paul Street, London EC2A 4JH. 01 377 6880.
Knight Games, Elektraglide: English Software, 3rd Floor, 1 North Parade, Parsonage Gardens, Manchester M60 1BX. 061 835 1358.
Lord of the Rings: Melbourne House, Castle Yard House, Castle Yard, Richmond. 01 940 6060.
The Phantom of the Opera: John Ransley, 3 Berry Close, Telscombe Cliffs, East Sussex BN9 7DW. 07914 88360.
Back to the Future: Electric Dreams, 0703 229694.
Space Doubt, Causes of Chaos, Nu-Wave: CRL, CRL House, 9 King's Yard, Carpenter's Rd, London E15 2HD. 01 533 2918.
Max Headroom: Quicksilva, Liberty House, 222 Regent St, London W1. 01 439 0666.



Hard Lines

HARDLY HAS THE GLUE DRIED ON A row of C128s emerging from Commodore's production lines than the Commodore 128D hits the streets. Launched at the Which? Computer Show in Birmingham in January, the C128D is a standard Commodore 128K multi-processor micro-computer with an integral 1571 disk drive.

According to Commodore, the package is aimed at the business user looking for a competitively priced entry point into computing. It features a single industry standard $5\frac{1}{4}$ inch double-sided disk drive built into the processor unit and

a detachable full travel professional keyboard. The package includes a mono monitor, with 40 and 80 column capabilities and costs £499 plus VAT.

Commodore UK's sales and marketing director, Chris Kaday said: "With immediate access to thousands of CP/M business packages, the software base for the 128D is formidable. The machine offers a high degree of functionally at a very competitive price."

There is also available a range of business applications for the machine's 128 mode including Script 128 and Micro Clerk. It's also compatible with C64 software. The 128D has a built in 6502 microprocessor, 2K RAM, 32K ROM, inbuilt DOS and is capable of fast data transfer rates.

In CP/M mode, the 1571 is fully compatible with numerous formats including Kay Pro, Osborne, IBM VER 1, IBM VER 2 and IBM CP/M 86. In this mode it offers 410K formatted capacity.

If you're thinking about buying a monitor then you'll be interested to hear that Philips Consumer Electronics is offering a free computer cable with every Philips colour monitor. When you buy your monitor you just fill in a pre-paid reply card with details of your computer and monitor. The relevant computer cable will be delivered in 24 hours by Videk Ltd, manufacturer of connecting cables. There are 35 cables on the range and prices range from £4 to £12. But hurry the offer closes in March.

Touch Line

Commodore (UK) Ltd: 1 Hunters Rd, Weldon, Corby, Northants N17 1QX. 0586 205555.

In Touch

Micronet has launched its own Sunday supplement, designed to rival Fleet Street's big boys.

Micronet officials report that a typical Sunday offering on the Net will feature a cult Science Fiction Soap opera entitled Macronet, gossip from an anonymous source in the computer industry, film and book reviews, news on media events, What's on in London (sorry the rest of the country) and an occasional celebrity chatline.

Sid Smith, Micronet's number two man, said: "The Sunday supplement reflects our determination to move away from exclusively computer oriented features into more general entertainment services. It should also bring in some very useful overtime but please don't quote me on that!" Sorry, Sid.

Touch Line

Micronet 800: 8 Herbal, London EC1R 5EJ. 01 278 3143.

Generally Speaking

HACKERS BEWARE, THERE IS NOW A new Terminal Security Device aimed at protecting dial-up minicomputer databases from unauthorised visitors.

The device, manufactured by Black Box, incorporates a hidden maze password protection scheme that will store up to 25 different passwords. Three attempts can be made to enter a correct password before the unit blocks all further tries for 15 minutes.

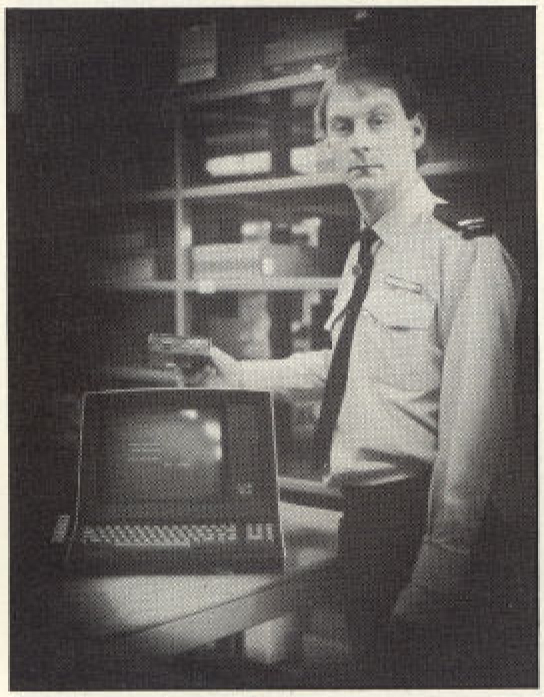
There is no prompt for a password but the computer is alerted if unauthorised entry is attempted.

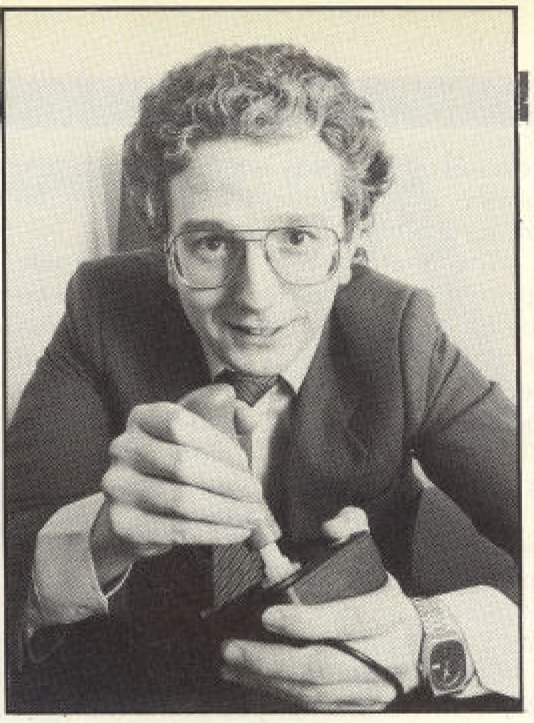
Three Commodore shows are to be hosted this year by Database Publications. The first of these is the official Commodore Show at the Novotel, London on May 31 to June 1. Commodore's Chris Kaday said: "We are looking forward to a long and fruitful association with Database. These are important times for us at Commodore with the launch of our new Amiga as well as maintaining the increasing interest in the 64 and 128.

The Two Commodore Horizons Shows will be held at UMIST in September and the Novotel in November.

The computer games industry has launched an appeal to aid the current government anti-drugs campaign. The appeal was launched at the InDin, the annual dinner for the computer games industry.

A spontaneous auction for champagne raised £14,000 to start the appeal. Companies who made the joint bid were Activision, Beyond, CRL, Computer Trade Weekly, Epyx, Melbourne House, Mikro-Gen, Mirrorsoft, Ocean, US Gold, Firebird, Geoff's Records, Gremlin Graphics and Imagine.





It is hoped that most funds will be raised from sales of a gmes compilation tape. The idea was conceived by Rod Cousens, a leading light behind the Soft Aid tape for the Ethiopian famine victims. The compilation tape will be called Off the Hook and will be available in the spring at a cost of £6.99. Companies who have already agreed to donate games are Activision, Beyond, Elite, Firebird, Gremlin Graphics, Melbourne House, Ocean and US Gold.

Touch Line

Database Publications: Europa House, 68 Chester Rd, Hazel Grove, Stockport SK7 5NY. 061 456 8383.
Black Box: PO Box 80, Reading, Berks RG2 0PS. 0734 866800.

Computer Weekends

Buying a computer is like buying a car, once you've bought it you have to spend quite a lot of time learning how to use it. Where cars are concerned there are plenty of schools which are willing to teach you enough to pass your driving test but with a computer you're on your own. Ardmore Adventure has decided to do something about this by providing weekend courses at Crest Hotels up and down the country.

In recent months, Ardmore has run courses, organised by Commodore (UK), which have proved so successful that Ardmore has now decided to continue the courses for anyone who wants to learn more about their Commodore computers. Families get extra benefits from these weekends because three children are given free accommodation when accompanied by two adults, under Crest's Welcome Break scheme.

The price of the course is only £30 per person with a charge of £27.50 per night for dinner bed and breakfast per adult. Further information is available from Ardmore Adventure Ltd, 23 Ramillies Place, London W1. Tel: David Walker 01-439 4461.

Commodore make relaxing in the sun up to £250 cheaper



Buy a Commodore 64 compendium 128, disk drive, printer or monitor now! Free inside each pack* you will find £250 worth of holiday discount vouchers.

Commodore your passport to pleasure

* Offer only applies to specially marked packs and is open to adults only. For full promotion terms and conditions see your Commodore dealer.



FREE £250 WORTH OF HOLIDAY DISCOUNT VOUCHERS

commodore

Commodore Business Machines UK Ltd., 1 Hunters Road, Weldon, CORBY, Northants. NN17 1QX
Available from: Boots, Clydesdale, Comet, Currys, Dixons, F. W. Woolworth, Granada, Greens, Harrods, John Lewis, John Menzies, Joplings Laskys, Rumbelows, Savacentre, Supreme, Ultimate, Vallances, W. H. Smith & Sons, Wigfalls, and all good computer stores.

Missives

The page where you tell us what you think about life, the universe but especially the world of Commodore computers.

In Memorium of Home Computing

The more I read in the computer press, the more I realise that home computing is slowly, but surely, dying. You'll probably think I'm talking rubbish, but let us consider the main causes of this situation.

First we have the computer manufacturers. Due to market pressures, manufacturers have shifted emphasis, first to games and more recently to more serious applications. Only a few years ago, there were a range of machines which were extremely well suited to home computing in its fullest sense. Such machines as the Nascom, Tangerine and Acorn Atom provided limited facilities in their basic form but offered immense potential for expansion. Despite the efforts of manufacturers, I refuse to accept that the new generation of machines will be of value to the home user. The average home user does not use a computer for accounts, data bases or word processing and the power of the 16 bit machines would be underused. I don't see the Amiga or ST being in the budget of the home user and, as older eight bit machines vanish, what will be left behind?

Next, we have the software houses. In the days of the Vic, ZX-81 and the early days of the 64 and Spectrum, we had a marvellous explosion in the software market with volumes of games etc. appearing. We did have indications of doom in the behaviour of such companies as Imagine but on the whole, it was a time of small adventurous companies. Unfortunately, money killed this scene and the big boys prevailed at the expense of the small trail blazers. Now we have expensive games written by armies of programmers, graphic artists etc. which, in truth, aren't always better than the earlier material. The only rays of sunshine in this area are the budget software houses which have shown us what can be done for only a couple of pounds per game.

Finally, we have the magazines. Due to the need for advertising revenue, magazines have been forced to follow the market trends with a resulting increase in emphasis on software reviews etc. The content of Practical Computing, for example, has no relevance to the magazines title or the material which appeared in its early days. Overall, there are very few magazines which cater for the home programmer. (Commodore User is probably the best example of a "user" magazine going down the drain).

Overall, the only option available to the home user, is the local user group. Only through such groups, or the parent group ICPUG, is there a flow of information, ideas or help to the computer enthusiast.

In conclusion, I fear that what promised to be the most exciting development of the last few decades may well simply prove to be a five year wonder. What do you think?

Allen Webb, Southport

Not so Easy Entry

I am having trouble with the Easy Entry program which appeared in the January issue of Your Commodore. The variables "check" and "VFY" never seem to agree with each other and consequently, I am not getting past the Error stage.

I by passed this stage and went on to the save portion of the program. But on entering "Mon-one" as a filename, 33280 as a start address and 33496 as the end address, I get an "Illegal quantity in line 440" message.

As "Sid" is poked in lines 540 and 550, presumably it should"burp" when an error is registered.

Also, no mention was made in the text that single and double digit numbers should be "padded" with preceding zeros on entry.

The overall principle of the program is quite good and it would be a godsend if I could get it working. However, it needs a few modifications to be up to Your Commodore standards.

I do hope that it is only a minor hiccup in the printing that line 400 is the first on the program and the start line 100 is somewhere near the middle!

Like Eric Pickering of Corwen, I too was a Y64 subscriber and became dissatisfied with its subsequent format. Letters to the editor were to no avail. I have therefore decided that the best remedy for any magazine which falls short of expectations is simply not to buy it!

A J Wiseman, Darlington

A few people do seem to be having problems with the Easy Entry program therefore it is worth going through the main problems here.

1) I'm afraid that two halves of the listing were printed the wrong way round the first time Easy Entry was printed. This does not affect the program in any way if you use the same line numbers, the computer will sort them into the correct order.

2) When you type in a line you must enter **all** *digits.*

3) A number of people are getting a syntax error in line 400 this is because they are typing the keyword TO instead of the Basic variable T0 (T,ZERO). Keywords cannot be used as Basic variables hence the error.

4) We have had a few problems with garbage collection. When you enter a string on the C64 it is stored in memory. When you enter a new string with the same name the old string is not always erased. Thus the old strings build up in memory and it is possible that the strings may overwrite the program. Adding the following line to the Easy Entry program will prevent this ever happening.

```
225 G=FRE (0)
```

5) Some people are having problems with the save routine. Once you save or load a section of code to or from tape or disk the program will return to the 'START ADDRESS' prompt. This is asking for the address at which you wish to continue typing. If the program does return to this prompt then it will have SAVEd or LOADed correctly. Don't forget that when you SAVE you must SAVE the program from the very beginning. You will not be able to list any of the data that you have saved, if you load it in and type LIST then you will only get a lot of rubbish printed to the screen. This is because you can't list machine code.

There are no other errors in the program apart from the one mentioned in 4) above. If you do get any other problems then I would advise that you check your typing very carefully.

Gemini & CentreSoft PRESENT

OFFICE MATE

With over four years experience of providing practical software solutions for business and home applications, Gemini have put together a selected range of famous titles for the Commodore 64 and 128 in TWO special packs, at VERY SPECIAL prices. These super value packs contain all the serious application software you're ever likely to need for your CBM, from word processing and database management to a complete professional business accounting system. Gemini's 'OFFICE MATE' and 'OFFICE MASTER' are here now - put that computer to work!

OFFICE MASTER



Word Processor

A fully-featured word processor program with text formatting - right justification - adjustable page length - text centering - output of all or part of text to printer - selective saving - file concatenation - block move - block delete - reverse print - graphics print - double width print - sorting - left and right margin selectable - word count - search and replace.

Mailing List

This program will enable you to keep a record of names and addresses and then print, examine, sort and find them, all with special selection techniques. Featuring the famous Gemini 'search-key' system, you have the option of creating your own dedicated coding routines for each name on the file.
For example, on most mail systems you are only given the standard headings such as name, street, town, county, etc. but with our system, you could for example find all companies listed that have a turnover in excess of a certain figure, or all subscribers who are behind with their subscriptions, or all people eligible for a Christmas card this year! A full range of utilities is callable from the menu including, ofcourse, label printing.

Database

Superior file management system with features found only with packages costing much, much more. Completely user-definable data entry format - colours definable - advanced mathematics using built-in machine code expression evaluator - fast sort on numeric and string fields - extensive searching with 'wild card' capability - user-definable data summaries - simple on-screen editing. A really POWERFUL database!

Home Accounts

Designed as a complete home accounting package, this program allows the user to set up and maintain a budget for items of household expenditure and compare actual with budget ,either numerically, or with the aid of chart graphics. A complete bank account routine is included, together with standard expenditure categories which may be changed to suit.

- Word Processor
- Database
- Mailing List
- Home Accounts
- Extensive Documentation

OFFICE MATE £12 Cassette or 1541 Disk **£15**

Step up to OFFICE MATE and save money! Usual price individually £79.80



- Database
- Mailist
- Word Processor
- Stock Control
- Cash Book
- Final Accounts
- VAT File
- Full Documentation

Database, Mailist and Word Processor as 'Office Mate' PLUS:
Cash Book / Final Accounts / VAT File.
Gemini's legendary cash book system for the CBM is a complete 'stand-alone' accounting software package, already in extensive use by both accountants and their clients.

General System Overview

The Gemini cash book package for the Commodore 64/128 microcomputer is designed for a hardware system consisting of:

1. Commodore 64/128 microcomputer.
2. 80 column printer.
3. Cassette or disk data storage.

Please note that running the program on disk will not change the way that the program works, but you will have the benefit of far greater speed and reliability for the loading and saving of files that a disk system provides.
There is a total of 199 nominal accounts, a large number of which may be defined by the user. You may have up to four cash control accounts, six bank control accounts, one sales ledger and one purchase ledger control account.
The program will store a data file consisting of:

1. The account titles.
2. The current cumulative balance on each account (debit or credit).
3. The net movement on each account for every month of the year.
4. VAT net sales and net payments figures, which are automatically created and maintained by the program.

This same data file is used by the FINAL ACCOUNTS program. The 'VAT FILE' which accompanies this package is designed primarily for those users on the Retailers special VAT schemes.
The main features of the CASH BOOK program are as follows:

1. Double entry routines for transactions through the cash/bank accounts and sales/purchase ledger control accounts.
2. Journal facility for the initial set up of accounts, or for adjustments to any of the accounts.
3. The facility to produce the following screened or printed reports:
   (a) Listing of all the nominal account titles.
   (b) Monthly transaction summaries.
   (c) A trial balance whenever required.
   (d) Screen VAT memo account balances (sales/net purchases, and VAT accounts).
   (e) A batch printing facility which provides details of all the transactions entered in the current run of the program.
4. The facility to extract regular management information such as cash/bank balances, debtors and creditors, sales, overheads, etc.
5. The program interfaces with the Gemini FINAL - ACCOUNTS program to enable Trading and Profit and Loss accounts and Balance Sheet to be produced whenever required. Comparative or budget figures can be shown alongside the actual figures using this program.
6. Screen prompts throughout the program to facilitate ease of use.
7. Storage of VAT information to assist in the preparation of periodic VAT returns.
8. Error trapping routines to minimise input errors.
9. The facility to handle the financial transactions of sole traders, partnerships, limited companies, clubs, etc.

Users registered for VAT are reminded that it is a statutory requirement to inform their local VAT office when they change their accounting records on to a new computerised accounting system.

OFFICE MASTER £25 Cassette or 1541 Disk

Dealers please note that 'Office Mate' and 'Office Master' now available also on BBC, Electron, and Spectrum.

MAIL ORDER TO:

Gemini

Gemini Marketing Limited, 'Gemini House', Dinan Way Trading Estate, Exmouth, EX8 4RS.

Please send me ______ CBM 64/128 'Office Mate' packs @ £12 (Disk £15)

Please send me ______ CBM 64/128 'Office Master' packs @ £25 Cassette / Disk (please delete as necessary)

Cheque / P.O.s enclosed to value £ ______
or please debit my Access/Amex No. ______

Name: ______

Address: ______

Signature: ______

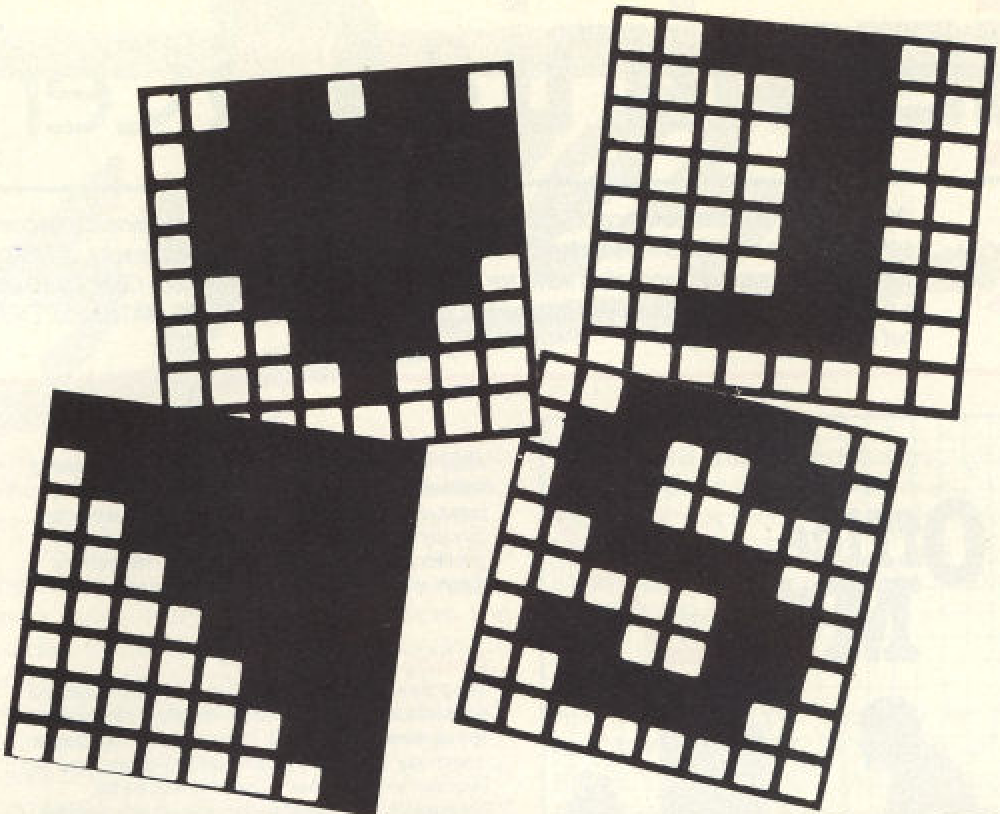
24 HOUR CREDIT CARD HOT LINE (0395) 265165 (4 lines)

Trade and overseas enquiries welcome

Sole distributors to the trade:

CentreSoft Ltd. Tel. 021-359-3020

Listings will be much easier to enter with our new system.



LISTINGS

COMMODORE LISTINGS ARE RATHER well known for the horrible little black blobs that always abound. Unfortunately the graphics characters which are used to represent graphic and control characters do not reproduce very well and they are also difficult to find on the Commodore keyboard.

In future all control and graphics commands will be replaced by a mnemonic within square brackets. This mnemonic is not typed out as printed in the magazine but rather the corresponding key or keys on the keyboard are pressed. For example [RIGHT] means press the cursor right key, you do not type in [RIGHT]. All of the keywords, what keys to press and how they are shown on the screen are shown below.

Any character that is accessed by pressing shift and a letter will be printed as [Sletter].

[SA] shift and A
[S+] shift and +

Any character that is accessed by pressing the Commodore key and a letter will be printed as [Cletter]

[CA] Commodore and A
[C+] Commodore and +
[C1] Commodore and 1

If any characters are repeated the mnemonic will be followed by a number. This number is how many times you should enter the character. Any number of spaces over one will also be represented in this form

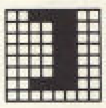
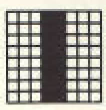
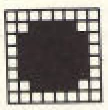
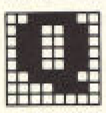
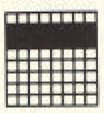
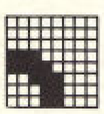
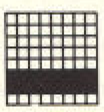
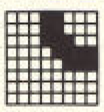
[RIGHT10] press cursor right 10 times
[C+10] press Commodore and + 10 times
[SPC10] Press the space bar 10 times

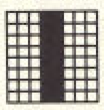
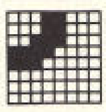
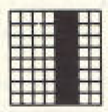
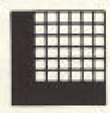
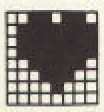
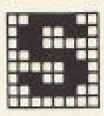
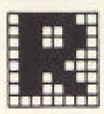
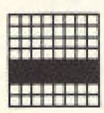
Any other characters should be easily recognisable for example CTRL-N means press CTRL and N and LEFT-ARROW means press the left arrow.

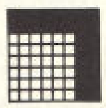
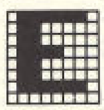
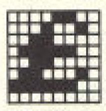
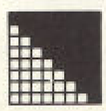
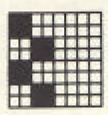
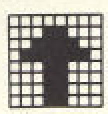
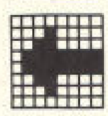
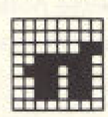
Any number of mnemonics can be enclosed in brackets for example

[SA10,SPC10,SA10]

means type 10 shift A's 10 spaces and another 10 shift A's.

Mnemonic	Symbol	what to press
[RIGHT]		left/right
[LEFT]		shift left/right
[UP]		Shift & up /down
[DOWN]		up/down
[F1]		f1
[F2]		shift & f1
[F3]		f3
[F4]		shift & f3

Mnemonic	Symbol	what to press
[F5]		f5
[F6]		shift & f5
[F7]		f7
[F8]		shift & f7
[CLEAR]		shift & CLR /HOME
[HOME]		CLR/HOME
[RVSON]		CTRL & 9
[RVSOFF]		CTRL & 0

Mnemonic	Symbol	what to press
[BLACK]		CTRL & 1
[WHITE]		CTRL & 2
[RED]		CTRL & 3
[CYAN]		CTRL & 4
[PURPLE]		CTRL & 5
[GREEN]		CTRL & 6
[BLUE]		CTRL & 7
[YELLOW]		CTRL & 8



turbo esprit
DURELL
software getting harder
DS 1
With Technical Assistance from Lotus Cars Ltd.
ARMED VAN SIGHTED ON EAST 22nd
turbo esprit
DURELL
BY MIKE RICHARDSON.
HIT-CAR CLOSING IN!
NEW YORK MAP
Spectrum and Amstrad
R.R.P. £8.95
DURELL sales dept.
Castle Lodge, Castle Green, Taunton TA1 4AB
DURELL COMPETITION
Entry Coupon
Cut out this coupon and send it with a completed entry form included with any of the new Durell games.

Eric Doyle has been examining Gremlin Graphics' range of C-16 games.

C16 — by Gremlin Graphics

OUR BELOVED EDITOR MAY HAVE gnomes at the bottom of his garden, but for the past month I've had gremlins in my C-16. Not the sort that gum up the works, but the current output of C-16 games from Gremlin Graphics.

The first is a not-to-be-missed bargain package containing four of Gremlin's earlier C-16 games for a mere £9.95. The games are: Dork's Dilemma, Xargon Wars, Petals of Doom, and Tycoon Tex.

Dork's Dilemma is to collect the parts of his spacecraft and reassemble them correctly. The pieces are spread throughout rooms which form a five by five grid. Each room has one or more exits and in the centre of the grid is a force field which holds a portion of the craft. Roaming around the passageways in each room are four nasty little Zobwats who will track you down and home in for the kill.

Dork looks a bit like one of those kids' toys that will wobble but won't fall over. His head and neck are duck-like but when he moves he pulls in his head and rolls everywhere. His armoury consists of an unlimited supply of time bombs with short fuses so when he sets one off he has to roll like mad to escape before it explodes.

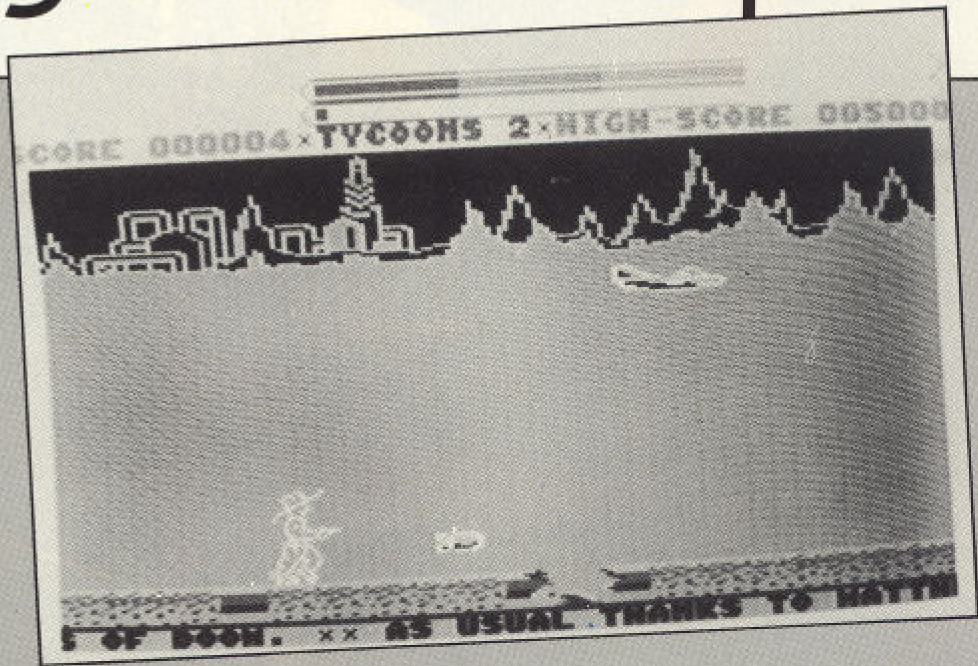
The unfriendly aliens can be eliminated by using the bombs sensibly, always bearing in mind that the blast does not penetrate the walls of the passages. After a number of these Zabwats have been blasted the piece is released from the force field and displayed on a grid at the right hand side of the screen.

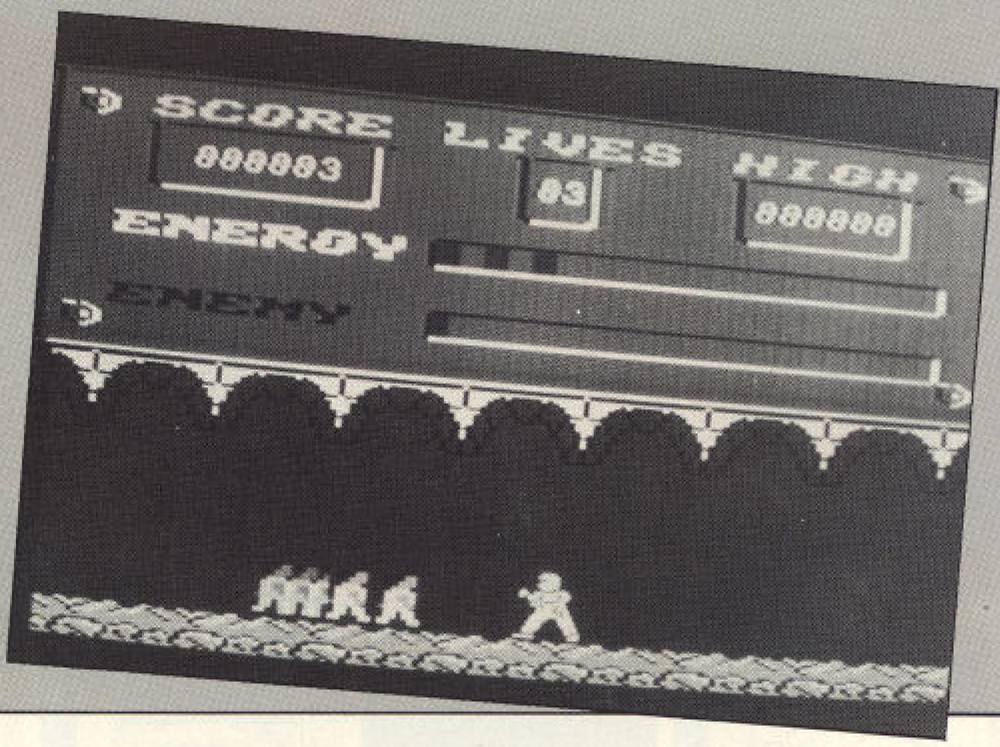
When Dork has collected enough pieces, the game can be paused while you sort them into the correct order. The game ends when the craft is completed.

Dork's Dilemma is a difficult game to play and some of the screens require very precise timing if he is to succeed in his mission.

Though Tycoon Tex is one of the visually more sophisticated games, I found it to be fairly tedious to play. Tex is a JR-type oil baron protecting his pipeline from attack by using his trusty six-shooter. As in all the best Western-style stories the 'six-gun' has an unlimited supply of bullets.

Firing as he goes Tex has to run the length of the pipeline jumping over the many breaks and avoiding the variety of missiles which assail him. At first the only problem is an aeroplane which drops bombs on to the pipe (or Tex if he gets in the way). Later screens bring the extra problems of bouncing bombs, crash explosives and there must be Indians in them thar hills because arrows keep flying across the screen.





The animation of Tex is very good indeed as he runs, jumps and crouches but I found that the length of the pipeline caused me a little eyestrain after a while which led to carelessness, then frustration and eventually tedium. These feelings will certainly not be shared by everyone who tries this game, but I gave up and I don't care if I never see this one again.

Xargon Wars is a game akin to Phoenix, the old arcade favourite, and despite the changes in arcade games over the years, this kind of shoot-em-up is still one of my favourite pastimes. You control the attack ship at the bottom of the screen and blast the hordes of aliens who appear above you, dropping bombs like raindrops from a stormy sky.

The aliens appear in a multitude of guises and formations, some in dense groups, others in long never-ending streams and still more in swirling snowstorms. Dodging whilst firing is the order of the day and the rather sensitive movement controls only add to the problem. With 16 screens to pass through, the Xargons kept me busy for many a happy hour and I still only reached screen five!

Gardening in outer space makes Percy Thrower's constant battle against greenfly and mildew looks like picnic. In Petals of Doom our green-fingered spaceman has to laser blast a host of alien blights if he is to grow a harvestable crop.

Unlike Percy, our hi-tech horticulturist has a radar screen to warn him of impending attack as he patrols his allotment in the sky. On this screen he can see from the mode of approach which aliens are attacking and work out from experience the best way to deal with them.

In the low gravity of his garden, our astronaut can use his jetpack to reach the high-flying threats but beware because his power pack can overheat if too much use is made of the gun or the jets. If this happens, and it will, his only recourse is to hide while his pack cools down and then try to undo the harm that will have been done to his crops.

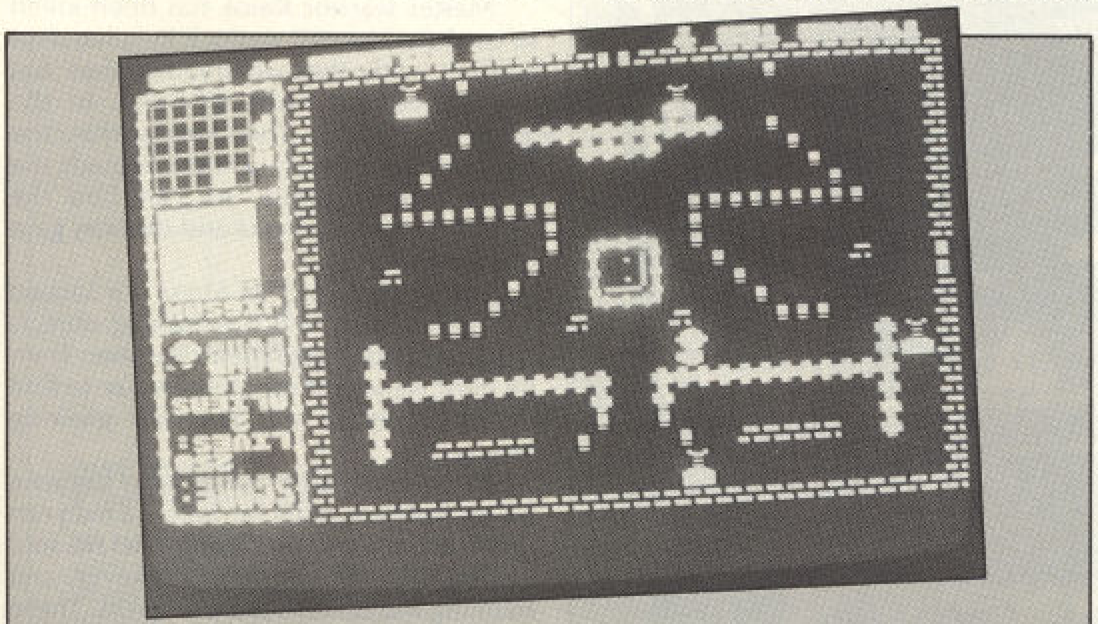
If all of the plants successfully reach maturity, there is always another allotment waiting just around the next screen and be careful because the attackers are even more numerous and ferocious.

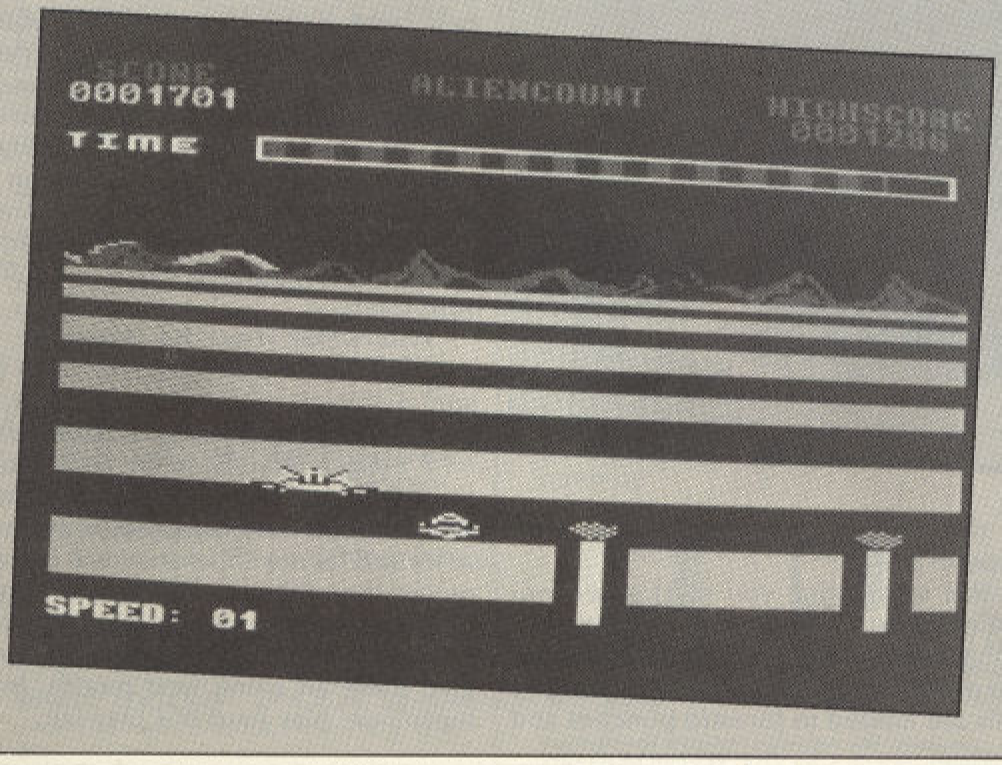
Petals of Doom is a game with limited appeal. After a while its repetitive nature tends to give way to boredom despite all of the frenzied activity.

Overall, this twin cassette pack offers a good variety of games for your money and I cannot honestly say that I would condemn any of them to being a waste of time or energy. The complaints that I have merely display my own tastes in computer games and another reviewer might see them in a totally different light.

In the Gremlin library there are four more games which are individually packaged at £6.95.

Xargon's Revenge picks up the story where Xargon Wars left off and for my money it shouldn't have bothered. If your idea of fun is to have wave after wave of attacking ships approaching your battle cruiser from screen right as you duck and weave at the right hand side, then this could be for you.





This game truly lacks variety of any kind except in the range of alien shapes attacking you. Control is limited to up and down movements as you fire continuously at the onrushing foe. In many similar games you can at least increase and decrease your speed to dodge the oncoming hordes and without this facility the game becomes a dull passive experience.

The only break to this rather uninspiring battle is the occasional appearance of a fuel pod which refills your dwindling supplies but I defy anyone to stay awake long enough.

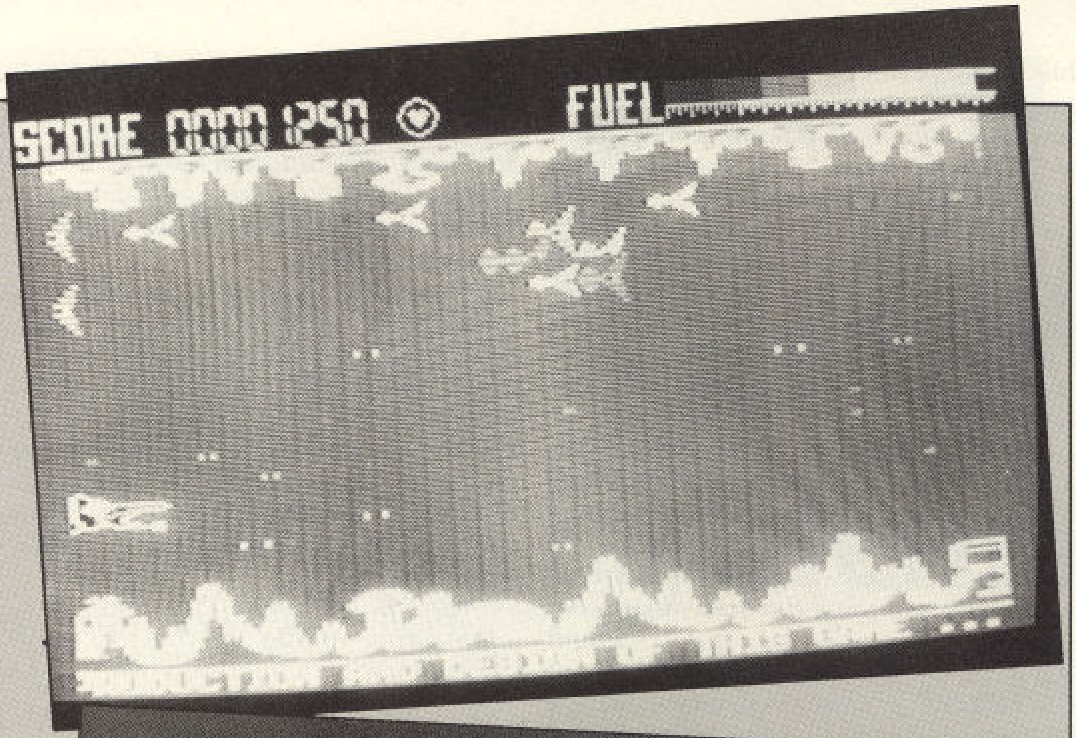
Another irksome feature of this game became obvious when I tried to play it using the keyboard option. To Gremlin's credit, all of their games have the option to use either a joystick or the keyboard keys but they fail to give the option to redefine the keys for the benefit of left-handed players. This is not my complaint in this case. The keyboard option invariably throws you into pause mode at the height of the action. An on-screen message implores you to press G to continue but on doing this the game aborts back to the beginning. I got nowhere in this mode and I feel that such an obvious failing ought to have been obvious during the play testing stage of the games development.

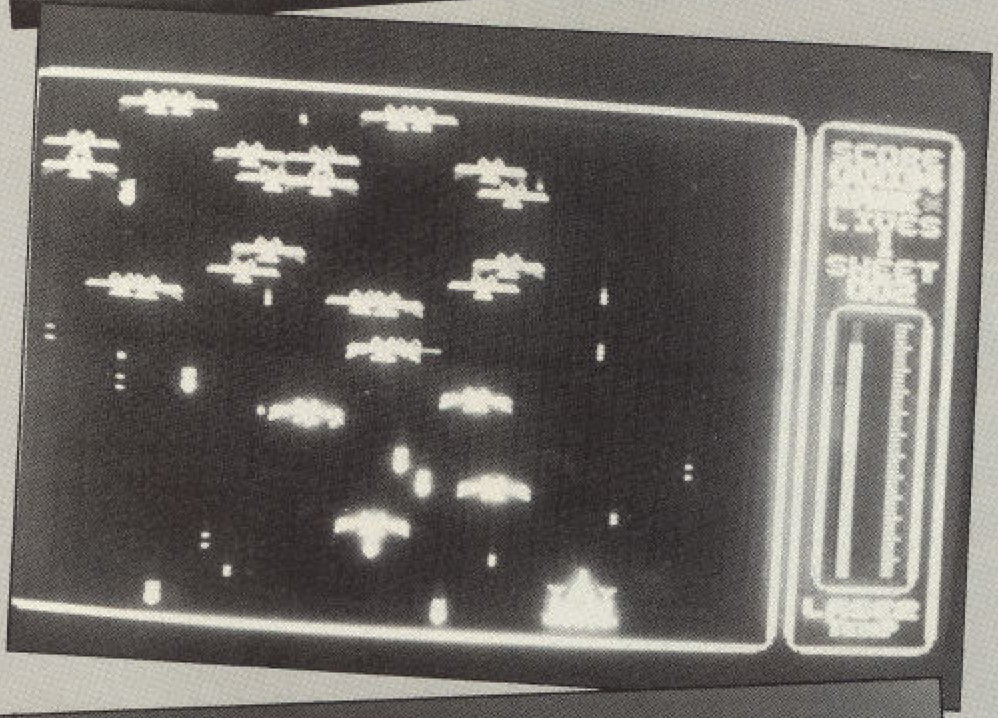
Gullwing Falcon is a different kettle of fish altogether. A Buck Roger's look-alike, it features pseudo 3D as the Falcon flies into the screen across a landscape called the Desert of Pillars. The pillars are immense gateways of stone through which the Falcon must steer to gain maximum points.

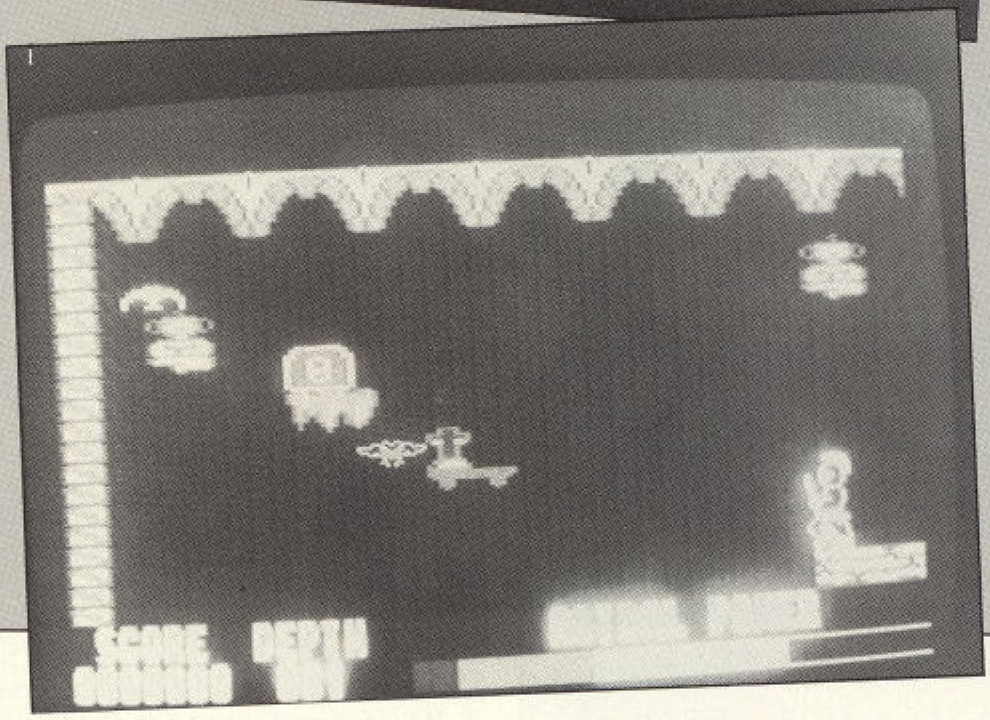
This game boasts 99 levels which means 99 waves of enemy ships which must be eliminated. Thankfully, Gremlin don't seem to go for ream after ream of imaginative, scene-setting drivel, just enough is given to set the scene and nothing more. However, sometimes the text can be a little devious. In this case you are advised to attempt to steer through the pillars because 'the pirates' ships dare not travel between the giant stone posts. What they don't tell you is that the pirates are not averse to hovering behind the posts waiting to pounce when you emerge! Take my warning, always blast like crazy as you pass between the posts.

Your forward speed can be controlled from interestingly fast Warp one to absolutely suicidal speeds. Quick reactions are needed to dispense with the aliens who may use flying saucers of various shapes or land hopping craft.

With so many screens on offer I defy anyone to complete this game.







Ah so! We come to the penultimate game on my list. Set in the land of Emperor To-Ming's Kung-Fu guards, you play the role of the banished Kung-Fu Kid. Whatever you did to deserve such a fate is not mentioned but you are now a lowly level eight fighter and you must progress to the highest level to regain your Emperor's trust.

This means that you must enter into battle against the Emperor's minions with each screen culminating in a battle against one of the Guardian Lizards. Your kung-fu skills are limited to forward punches and kicks which must be used against attackers who come from behind as well as in front. Colossal leaping skills are also yours but in battle try to conserve your energy for the final fight with the lizard.

Sharigan stars zip across the screen and you must punch these away from the second screen onwards, with swords, daggers and magical pots adding to the excitement of higher levels.

The graphics are adequate and realistically animated and the timing for your blows must be carefully practised if you don't want the stuffing knocked out of you before the end of the first screen. Altogether Kung-Fu Kid is an enjoyable little romp but limited in its staying power I fear.

Keeping the best to last we reach Sword of Destiny which I feel is the jewel in Gremlin's C-16 crown. This is a megamaze game which I really feel is in a totally different class from any of the other games. Instead of merely fighting your way through each single screen this maze requires you to cross and recross the same screens in your search for the Sword of Destiny.

Master warrior Kelok has been killed in battle by the evil warlock Xorphas. Not content with this victory, Xorphas has imprisoned the heart of Kelok in the Abyss of Death. This means that the warrior's soul must wander through the Abyss forever unless he can find the Sword to free his heart and thereby gain eternal peace.

The tunnels of the Abyss are locked and Kelok must find the flashing objects which will open each new chamber. Then he must wander through the maze to find the new region and the next glowing artefact.

Being dead has its advantages because falling holds no fears for Kelok, a man can only die once and he already has! His soul can be drained of energy however, and roaming the caverns are evil spirits. These guardians of death must be smitten with a bolt from Kelok's magic sword both to save and replenish his dwindling energy.

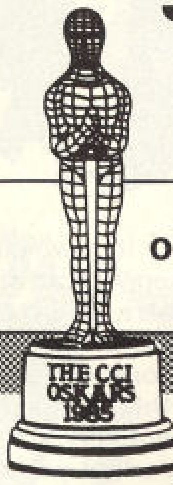
Occasionally, Kelok will find himself at the foot of a shaft with no obvious exit. In this case he must mount the rising air currents which issue from vents in the ground. Staying on the air current is no easy task because the death's guardians will knock him off if he carelessly collides with them.

Will Kelok free his heart or will Xorphes' awful plan succeed? Only time and skilful play will tell. Absolutely, the best of the Gremlin Graphic's bunch.

I believe that Gremlin and Mastertronic have the best C-16 collections available, anyone who disagrees is invited to write and let me know and I will certainly review their suggestions. In the meantime I have been collecting the best of the rest C-16 programs currently available and I'll give these an airing next month. In the meantime, play hard but play fair.

THE FINAL CARTRIDGE

THE FIRST OUTSIDE OPERATING SYSTEM FOR THE CBM 64 *



UTILITY OF THE YEAR 1985



£45

This new operating system built in a cartridge does not use any memory and is always there. Compatible with 98% of all programs.

DISK TURBO - 6 times faster loading - 8 times faster saving.

TAPE TURBO - 10 times faster, even with files - normal Commodore commands - compatible with standard turbo's.

ADVANCED CENTRONICS INTERFACE - compatible with all the well-known centronics printers and Commodore printer programs. Prints all the Commodore graphics and control codes (important for listings).

SCREEN DUMP FACILITIES - of low-res Hi-res and multicolour screens!! Prints full page with 12 shades of grey for multicolour pictures even from games and programs like Doodle, Koala pad, Printshop etc. Searches automaticly for the memory-address of the Picture. Special version available for the CBM 801 and 803 printers.

24K EXTRA RAM FOR BASIC-PROGRAMS AVAILABLE: Two new commands ,,Memory read'', ,,Memory write''. They move 192 bytes with machinelanguage-speed anywhere in the 64K Ram of the CBM 64. Can be used with strings and variables.

BASIC 4.0 COMMANDS - like Dload, Dsave, Dappend, Catalog, etc.

BASIC TOOLKIT - with Auto, Renum (incl. Goto and Gosub), Find, Help, Old, etc.

* works with C128 in the 64 mode.



Original multicolour full page screen dump print out.

PREPROGRAMMED FUNCTION KEYS: - Run, Load, Save, Catalog, Disk commands, List (removes all list-protections).

KEYBOARD EXTRA'S - Allows you to delete part of a line; stop and continues listings; move cursor to lower lefthand corner. Pokes and Syscalls in Hex. Type-command operates your printer as a typewriter.

COMFORTABLE EXTENDED ML. MONITOR: - with relocated load scrolling up and down. Bankswitching, etc. - **does not reside in memory.**

RESET SWITCH: - resets to monitor; resets with old, resets to Hi-res printing; resets every protected program.

ON/OFF SWITCH - we hope you never need that one.

FREEZE FRAME

Stops and continues almost every program and allows you to make a **total back up** to disk or tape automatically.

14 Days money back guarantee if you are dissatisfied.

12 Months replacement guarantee.

H&P (ome ersonal) COMPUTERS

U.K. ORDERS Available by the wellknown Commodore Dealers or directly from

H & P Computers
9 Hornbeamwalk
Witham Essex CM8 2 SZ England d.
Telephone: 0376 - 51 14 71.

DATA ERROR: LOCATE AND DESTROY

P. Green reveals some techniques to make it easier for you to find and correct data errors.

MOST OF YOU, AT SOME TIME OR other must have typed in a program with lots of Data statements and then experienced a checksum error when the program was run.

You are then faced with the job of going through all the Data statements to find the error or errors. This is a time consuming job and tends to get very boring. There are, however, a few short cuts that you can take when doing this which should make the task easier for you.

There are two types of error which can crop up in Data statements. Firstly, the number of data bytes can be incorrect. And secondly, there can be one or more faulty bytes of data. The first will invariably result in a checksum error as well.

These may seem like obvious statements, but these errors have different effects on the screen and require different techniques for error finding.

Basics

The first things to be checked are the READ and ERROR TRAP routines. Too many data bytes will only express itself as a checksum error if the READ routine counts the data bytes. In this case the error trap will inform you that there are too many data bytes.

Too few data bytes will express itself in the statement 'OUT OF DATA ERROR' if a FOR/TO/NEXT data entry routine is used. If a X=X+1/READA/IFA<OTHEN repeat type of routine is used, then it can be treated in the same way as too many data bytes. After a program has been run and the error trap has halted it, you should obtain as much information as possible before actually checking the data. Below is a simple error trap which I will use to explain the procedure.

```
10 FOR X=0T /O100:READA
20 POKE49152+X,A:CK=CK+A:NEXT
```

In order to get the required information, type the following in direct mode:

```
PRINT X,A,CK,PEEK(63)+PEEK(64)*256
```

'X' will tell you how many data bytes have been entered in the case of an 'OUT OF DATA ERROR'. 'A' will tell you the last data byte to be entered.

The difference between 'CK' and its correct value is very useful since if you make a note of it and amend it as you find errors, it will give you an indication of how many more errors there are.

If you examine the memory map of the C64, you will see that locations 63 and 64 will give you the current dataline number. This is useful since it will give you the last data line number to be entered.

Armed with this information you should have an idea of the number of data bytes too few or too many which you have, and also the extent of the checksum error.

Short Cuts

The ease with which you can make short cuts depends on whether the data is for machine code, sprites, user defined characters or screen memory. The following tips are for the C64 but should also work for the Vic 20.

Machine Code

I will start with machine code data which is the most difficult to handle. There are no short cuts to this one unless you possess at least one machine code monitor and have some knowledge of assembly language and the machine on which you are working. The method is to run the program, new it, and load your machine code monitor. If you know the area of memory into which the database was loaded, then you can disassemble the area to see if there are any obvious errors. A common place for machine code routines to be located is a $C000 or 49152. The success of this method will depend on your own experience.

The reason why I prefer to use more than one monitor is that if the m/c is at $C000 then you can use one that is located low in the memory map. If the m/c is low in the memory then you can use a monitor that is located high in the memory. Short m/c programs are sometimes located in the cassette buffer. Therefore you cannot run the program and then load the monitor if it is on tape. It will be necessary to use a monitor which is located high in the memory and then load and run the program.

Sprites

Sprite data is one of the easiest to check, especially if the sprites are not multicolour. The procedure is as follows:

1. Temporarily put a STOP command in the program to stop it even if the checksum is OK.
2. Run the program
3. Turn sprite # 0 on -POKE53269,1
4. Set sprite colour -POKE53267,0
5. Enlarge sprite -POKE53271,1
-POKE53277,1
6. Set co-ordinate -POKE53248,250
-POKE53249,200
7. Set sprite pointer -POKE2040,LOC

where LOC is the start location divided by 64. This can be obtained from the program.

The first sprite should now appear on the right side of the screen. I presume that you have a rough idea what the sprite should look like and should be able to spot any errors. All you have to do is make any amendments to the data and run the program again. Any changes in the data should alter the sprite. Once this sprite is OK, then increment the sprite pointer and the next sprite should be displayed and can be checked in the same way. This can be done for all the sprites.



Characters

Finding faults in user defined character data is nearly as easy as sprite data but may take a little more time and more understanding of what you are actually doing.

Before you start you should study the program to find out exactly where the data is to be stored (A common place to store character data starts at 12288), and what characters should be produced (are they just modified letters or completely new characters such as parts of a wall or a house?). You can also find the command which turns the new character set ON. This will look something like POKE53272,28. There may be another command to switch memory banks, but this will make things much more complicated and it probably is not worthwhile using short cuts in this case. Then follow these instructions:

1. Temporarily put a STOP command in the program to stop it even if the checksum is OK.
2. Run the program.
3. Turn new character set on: POKE53272, 28 is for a character set starting at 12288.
4. The screen may now appear to have funny characters on it, but don't panic. If you are in any doubt, clear the screen and return to the standard character set by entering POKE53272,21 and return.
5. Start typing the alphabet and note any characters that look wrong.
6. Return to the standard character set.
7. Since each character requires eight bytes, it is not difficult to find the faulty byte of data. For instance, if the letter 'D' looks wrong, then, refering to the handbook or reference guide, the data for 'D' starts at the 33rd byte of data. The formula is 1+8*POKE No.
8. After the data has been corrected and SAVEd, the procedure can be repeated from instruction 2 until all the errors have been found.

This is fine if the program is for a new character set, but what if the characters from part of a larger object such as enlarged characters or part of a wall or house? The procedure is then similar to that shown above except with the following modification (insert it instead of 5. above):

5. It is necessary for you to examine the listing again, this time to find out how the program prints the new characters. Usually two or more characters are printed together to form a larger object. This may be in a line or in a block e.g.:

```
                   abc
abc  orghij   or   defgh
                   hijk
```

You then need to print the characters in the same way for them to make sense.

Screen Memory

Screen memory, which usually starts at 1024, is often used to produce a title screen or a background picture. Screen memory data is the easiest to sort out. The method is as follows:

1. Temporarily put a STOP command on the program to stop it even if the checksum is OK.
2. Some models of the C64 require that colour memory, which starts at 55296, is POKEd with a character colour code. This can be done in direct mode – if your program does not do this for you before entering the data – if your program does not do this for you before entering the data – by typing in:

```
FORA=0TO999:POKE55296+A,1:NEXT
```

This will make all the POKEd characters on the screen white.

3. Place the cursor about half way up the screen and run the program. The screen should then begin to fill with the POKEd characters.
4. It should then be quite a simple matter for you to find any faulty data by examining the screen with the aid of the video memory map.

Final Word

I hope that these suggestions will make it easier for you to type in programs from magazines and books. These sources of software can provide quite a lot of enjoyment, both in the typing and running, and can, if you take the trouble to try and understand the programs, teach you quite a lot about the use of Basic.

Telephone EXCHANGE

K Otton and A Adams introduce you to the intricate world of computer communications.

THE AIM OF THIS ARTICLE IS to give a brief insight into the world of communication using a computer, a modem and a telephone line and to explain the features of the Terminal program, published in conjunction with this article.

Firstly, a modem is a piece of electronic equipment which is connected to the computer via an RS232 interface and attached to the telephone line via a jack plug. The name, is an abbreviation of modulator/demodulator. Its main function is to convert data coming from the computer into audio signals which are then transmitted, along with a carrier tone, down a conventional telephone line. The noises are similar to those heard if a computer tape is played back on a standard hi-fi system. Data noises coming from the far end work in reverse, sending data into the computer.

The RS232 is just an interface which allows the modem to remain standard for all computers. It is only the connections which vary slightly. At this stage it is probably a good idea. to examine the cost of setting up such an operation. Modems vary in price from approximately £25 to over £200 depending on the facilities offered. On some of the cheaper versions an RS232 interface may also be required, the dearer ones tend to have this already built into the connecting lead and are much easier and tidier to hook up.

Before venturing into the land of bulletin boards, the cost of telephone calls must be taken into account. The average time on a bulletin board can vary between 15 and 40 minutes. Some boards will set a time limit – usually between 20 and 90 minutes – others have no limit. Time is something to be remembered when a telephone bill for £100 or so drops through the letter box (even more so if your parents are paying the bill!).

PROGRAM: BASIC MOVE

```
10 POKE 43,0:POKE 44,48
   :POKE 12287,0:NEW
```

PROGRAM: BOOT GENERATOR

```
1000 PRINT "[CLEAR,DOWN]WHEN
     RUN THIS PROGRAM WILL
     GENERATE AN"
1010 PRINT "AUTO BOOT PROGRA
    M CALLED"
1020 PRINT "[DOWN]COMMS BOOT"
1030 PRINT "[DOWN]TO BOOT
     THE COMMS PROGRAM IN FUT
    URE TYPE"
1040 PRINT "LOAD "+CHR$(34)+
    "COMMS BOOT"+CHR$(34)+",8,
    1"
1060 PRINT "[DOWN3,RVSON]
    SAVE[RVSOFF,SPC]
    THIS PROGRAM[SPC,RVSON]
    BEFORE[RVSOFF,SPC]YOU RUN
     IT AS"
1070 PRINT "IT WILL BE ERASE
    D WHEN FINISHED"
1080 PRINT "[DOWN3]ANY KEY
     TO CONTINUE"
1090 PRINT "RUN/STOP TO ABOR
    T"
1095 GET K$:IF K$="" THEN GO
    TO 1095
2000 FOR L=0 TO 5:CX=0
    :FOR D=0 TO 15:READ A
    :CX=CX+A:POKE 679+L*16+D,A
    :NEXT D
2010 READ A:IF A<>CX THEN PR
    INT"ERROR IN LINE";
    2040+(L*10):STOP
2020 NEXT L
2040 DATA 49,84,69,76,67,79,
    77,47,86,49,169,1,162,8,
    160,1,1184
2050 DATA 32,186,255,169,10,
    162,167,160,2,32,189,255,
    169,0,32,213,2033
2060 DATA 255,238,167,2,173,
    167,2,201,51,208,232,169,
    0,141,2,3,2011
2070 DATA 169,204,141,3,3,76,
    0,204,3,3,76,0,204,0,0,0,
    1086
2080 DATA 0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0
2090 DATA 0,0,0,0,0,0,0,0,0,
    139,227,186,2,0,0,255,809
3000 POKE 43,167:POKE 44,2
    :POKE 45,7:POKE 46,3:CLR
3040 SAVE "COMMS BOOT",8,1
3050 SYS 64738:REM NEW MACHI
    NE
```

PROGRAM: COMMS GEN1

```
2000 FOR L=0 TO 64:CX=0
    :FOR D=0 TO 15:READ A
    :CX=CX+A:POKE 2048+L*16+D,
    A:NEXT D
2010 READ A:IF A<>CX THEN PR
    INT"ERROR IN LINE";
    2040+(L*10):STOP
2020 NEXT L
2040 DATA 42,0,0,42,42,42,42,
    42,42,42,42,42,42,42,42,
    42,588
2050 DATA 42,42,42,42,42,42,
    42,42,42,42,42,42,42,42,
    42,42,672
2060 DATA 42,42,42,42,42,42,
    42,42,42,32,32,32,32,32,
    32,32,602
2070 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,32,32,
    32,32,512
2080 DATA 32,32,32,32,32,32,
    32,32,32,32,32,32,32,32,
    32,42,522
```

We have mentioned bulletin boards and this may be new to some of you, so a quick explanation is in order. Bulletin boards are normally set up and run by somebody interested in computers with enough spare time for the upkeep of the system. They can be either individuals or companies. The services they provide vary enormously, a few examples are: Sales and wants, technical help, messages, jokes, stories, uploading or downloading of programs. Also provided can be lists of other bulletin boards, telephone numbers and general news and interests. Some will also have special sections, such as scouting. Some specialise in one subject such as medicine, mail order stock etc. We shall come back to bulletin boards later but now let's take a look at what modems have to offer.

Baud Rates

The standard baud rates are usually 300/300, 1200/75 or 75/1200.

Auto Dialling

This is a fairly obvious one. The number to be called can usually be typed in on the computer keyboard. The autodial then takes over and pulses the number out. Some are able to redial if the number is busy or try several times before giving up.

Auto Answer

This will recognise the ringing coming in on the telephone line and send a signal to

a) start the computer program

```
2090 DATA 42,32,32,32,32,32,
   32,32,32,32,32,32,42,42,
   32,84,594
2100 DATA 5,18,13,9,14,1,12,
   32,42,42,32,32,32,32,32,
   32,380
2110 DATA 32,32,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,532
2120 DATA 32,32,32,32,32,32,
   32,45,45,45,45,45,45,45,
   45,32,616
2130 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2140 DATA 42,32,32,32,70,21,
   14,3,20,9,15,14,32,32,32,
   32,432
2150 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2160 DATA 32,75,5,25,32,32,
   32,42,42,32,32,32,32,32,
   32,32,541
2170 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2180 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2190 DATA 42,32,32,32,32,60,
   49,62,32,32,83,1,22,5,32,
   83,631
2200 DATA 3,18,5,5,14,46,46,
   46,46,46,46,46,46,46,46,
   46,551
2210 DATA 46,70,49,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,601
2220 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2230 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2240 DATA 42,32,32,32,32,60,
   50,62,32,32,86,9,5,23,32,
   83,644
2250 DATA 3,18,5,5,14,46,46,
   46,46,46,46,46,46,46,46,
   46,551
2260 DATA 46,70,50,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,602
2270 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2280 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2290 DATA 42,32,32,32,32,60,
   51,62,32,32,84,15,7,7,12,
   5,537
2300 DATA 32,70,9,12,5,32,79,
   78,46,46,46,46,46,46,46,
   46,685
2310 DATA 46,70,51,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,603
2320 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2330 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2340 DATA 42,32,32,32,32,60,
   52,62,32,32,84,15,7,7,12,
   5,538
2350 DATA 32,70,9,12,5,32,79,
   70,70,46,46,46,46,46,46,
   46,701
2360 DATA 46,70,52,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,604
2370 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2380 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2390 DATA 42,32,32,32,32,60,
   53,62,32,32,83,5,14,4,32,
   80,627
2400 DATA 18,5,45,84,25,16,5,
   4,32,77,5,19,19,1,7,5,367
2410 DATA 46,70,53,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,605
2420 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2430 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2440 DATA 42,32,32,32,32,60,
   54,62,32,32,83,1,22,5,32,
   70,623
2450 DATA 9,12,5,46,46,46,46
   46,46,46,46,46,46,46,46,
   46,624
2460 DATA 46,70,54,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,606
2470 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2480 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
```

Figure 1

```
MATRIX
/–/–/–/–/
/–/–/–/–/
/–/–/–/–/

NEXUS–1

     OPERATED BY:
MATRIX COMMUNICATION
     SYSTEMS

 SYSOP: KEN FARNEN

WELCOME TO THE MATRIX!

THIS IS NEXUS–! (MANCHESTER)

AUTHORISED USERS PLEASE LOGIN
WITH YOUR NAME' AS PROMPTED.

FIRST NAME? MIKE
LAST NAME? ROWE
SEARCHING USER FILE...
CALLING FROM LONDON
IS THIS CORRECT? Y
ENTER YOUR PASSWORD:  *****

MCS Welcomes MIKE ROWE
Calling from LONDON
Your last time on was 11/14/85  12:18
You have read through message 8295
Current last message is 8543
You have called this system 17 times
before
You are caller number 11399
You are authorised  40 minutes this call
Searching Message Base...
You have no personal messages waiting.


–Matrix–
=ROOT Menu=
(M)essages...Of all types
(O)n–Line  .. Magazine (Features and
Info.)
(N)ew!  .. Whats New on the Matrix
(E)ntertainment .. fun time!
(U)ser .. Micro users groups
(R)esults .. Results of the
Questionaire

(GT*!) .. System Functions:
(G=Goodbye   T=Time)
(*=Utilities   !=Help)
```

```
2490 DATA 42,32,32,32,32,60,
   55,62,32,32,83,1,22,5,32,
   83,637
2500 DATA 3,18,5,5,14,32,84,
   15,32,68,47,84,46,46,46,
   46,591
2510 DATA 46,70,55,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,607
2520 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
```

b) answer the telephone line and connect the computer.

When only one line is available for both normal calls and modem calls, a caller wishing only to speak will be answered by the modem's high-pitched tone and will be cut off after a set time if no incoming carrier is detected.

Self Test

Again, this one speaks for itself. Normally, this is a switch, push button or moveable link which connects the transmit and receive circuits together giving a loop back into the computer. This enables you to send data, receive the data you have sent and display it on your screen. This is a quick and easy way to test the program and the RS232 interface.

Indicator Lights

If you examine all the modems available, you will notice that some have several little indicator lights. These can have a variety of purposes, the most common being: power on, transmit and receive data, carrier OK and on line. Most of which are self explanatory.

The basic modem will simply offer 300/300 CCITT V21 standard, originate only. A typical British Telecom modem 13A offers these facilities, with the addition of self test. It can cost as little as £45 and is available with full instructions and fully tested from Distel Electronics. Also required is an RS232 interface available from Maplin Electronics.

The most common baud rate is 300/300. This refers to the speed at which data is transmitted and received and therefore 1200/1200 is four times faster.

The 1200/75 is a split baud rate where the board sends out at 1200 bits per second and waits for a reply which you send at 75 bits per second. The reverse would apply where 75/1200 is shown.

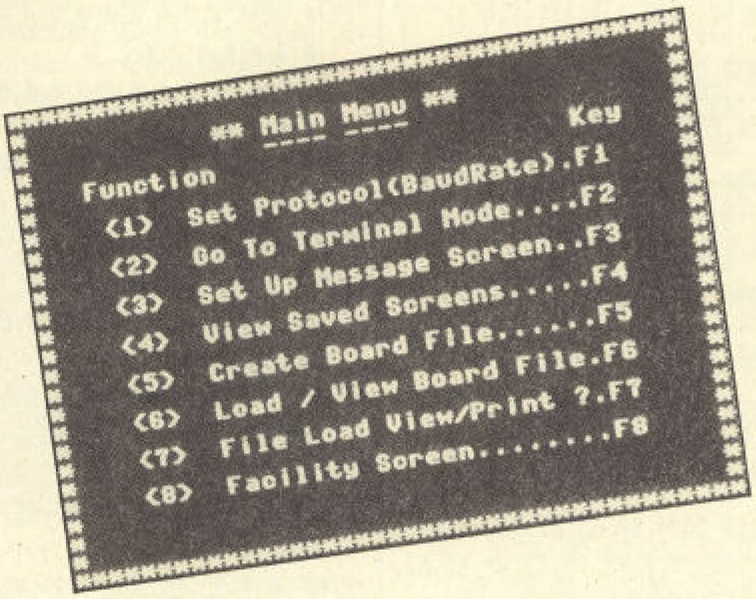
As the 1200/1200 baud rate is more prone to data corruption, the 75/1200 or 1200/75 rates are preferred. This also means that time can be saved (phone bill again) when receiving or transmitting data at the 1200 baud rate. Since the response from the receiver – the end sending at 75 – is

```
2530 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2540 DATA 42,32,32,32,32,60,
   56,62,32,32,69,24,9,20,32,
   20,586
2550 DATA 15,32,77,1,9,14,32,
   77,5,14,21,46,46,46,46,46,
   527
2560 DATA 46,70,56,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,608
2570 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2580 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2590 DATA 42,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,522
2600 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2610 DATA 32,32,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,532
2620 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2630 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2640 DATA 42,42,42,42,42,42,
   42,42,42,42,42,42,42,42,
   42,42,672
2650 DATA 42,42,42,42,42,42,
   42,42,42,42,42,42,42,42,
   42,42,672
2660 DATA 42,42,42,42,42,42,
   42,42,70,0,0,0,0,0,18,0,
   424
2670 DATA 70,0,0,0,0,0,18,0,
   70,0,0,0,0,0,18,0,176
2680 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
4000 PRINT"[CLEAR]NEW"
   :PRINT"[DOWN2]
   LOAD"+CHR$(34)+"COMMS GEN
   2"+CHR$(34)+",8"
4005 REM ** CHANGE ,
   8 IN ABOVE LINE TO ,1 IF
   YOU ARE USING TAPE **
4010 PRINT"[DOWN4]RUN"
4020 POKE 631,13:POKE 632,13
   :POKE 633,13:POKE 198,3
   :PRINT"[HOME]"
```

PROGRAM: COMMS GEN2

```
2000 FOR L=0 TO 80:CX=0
   :FOR D=0 TO 15:READ A
   :CX=CX+A:POKE 4096+L*16+D,
   A:NEXT D
2010 READ A:IF A<>CX THEN PR
   INT"ERROR IN LINE";
   2040+(L*10):STOP
2020 NEXT L
2040 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2050 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2060 DATA 32,32,32,32,32,32,
   32,32,32,2,15,1,18,4,19,
   32,379
2070 DATA 14,1,13,5,32,32,32,
   32,32,32,32,32,32,32,32,
   32,417
2080 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2090 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2100 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2110 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2120 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2130 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2140 DATA 32,20,5,12,5,32,14,
   15,32,32,32,32,32,32,32,
   32,391
2150 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2160 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2170 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2180 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2190 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2200 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2210 DATA 32,32,32,32,32,32,
   32,32,32,19,5,20,32,1,20,
   32,417
2220 DATA 32,32,51,48,48,47,
   56,47,49,32,51,12,9,14,5,
   47,580
2230 DATA 6,21,12,12,47,14,
   15,32,16,1,18,9,20,25,32,
   32,312
2240 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2250 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2260 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2270 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2280 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2290 DATA 32,77,1,3,18,15,32,
   83,5,20,32,65,20,45,32,32,
   512
2300 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2310 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2320 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2330 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2340 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2350 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2360 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2370 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2380 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2390 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
```

```
2400 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2410 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2420 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2430 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2440 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2450 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2460 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2470 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2480 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2490 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2500 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2510 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2520 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2530 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2540 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2550 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2560 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2570 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2580 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2590 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2600 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2610 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2620 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2630 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2640 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2650 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
```




```
2660 DATA 32,32,32,32,32,32,
   32,32,0,0,0,8,0,0,0,0,264
2670 DATA 0,0,0,0,0,0,144,0,
   0,0,0,0,0,0,0,0,144
2680 DATA 147,17,32,69,78,84,
   69,82,32,70,73,76,69,78,
   65,77,1118
2690 DATA 69,58,0,13,17,32,
   69,78,84,69,82,32,70,73,
   76,69,891
2700 DATA 78,65,77,69,58,0,
   147,17,17,32,32,69,82,82,
   79,82,986
2710 DATA 32,79,78,32,83,65,
   86,69,47,76,79,65,68,58,0,
   166,1083
2720 DATA 48,48,75,75,50,50,
   50,52,0,52,0,79,71,49,45,
   32,776
2730 DATA 191,48,48,75,75,50,
   50,50,52,0,52,0,79,71,49,
   45,935
2740 DATA 32,147,17,32,76,79,
   65,68,32,68,73,82,69,67,
   84,79,1070
2750 DATA 82,89,32,89,47,78,
   0,147,17,32,95,32,77,69,
   78,85,1049
2760 DATA 13,17,32,83,32,83,
   67,82,69,69,78,13,17,32,
   80,32,799
2770 DATA 80,82,73,78,84,69,
   82,0,13,13,32,32,32,32,32,
   32,766
2780 DATA 80,82,69,83,83,32,
   83,80,65,67,69,32,66,65,
   82,32,1070
2790 DATA 84,79,32,67,79,78,
   84,73,78,85,69,13,0,83,0,
   140,1044
2800 DATA 2,0,2,5,0,8,13,13,
   13,32,80,82,69,83,83,32,
   517
2810 DATA 65,78,13,32,75,69,
   89,0,36,48,1,0,0,0,0,0,506
2820 DATA 0,0,39,46,53,0,0,0,
   0,0,0,21,21,21,60,63,324
2830 DATA 66,0,69,21,21,21,0,
   21,76,86,0,97,0,109,0,121,
   708
2840 DATA 21,255,255,255,255,
   255,255,255,255,255,255,
   255,255,255,255,255,3846
4000 PRINT"[CLEAR]NEW"
   :PRINT"[DOWN2]
   LOAD"+CHR$(34)+"COMMS GEN
   3"+CHR$(34)+",8"
4005 REM ** CHANGE ,
   8 IN ABOVE LINE TO ,1 IF
    YOU ARE USING TAPE **
4010 PRINT"[DOWN4]RUN"
4020 POKE 631,13:POKE 632,13
   :POKE 633,13:POKE 198,3
   :PRINT"[HOME]"
```

normally only a single key selection, even 75 baud rate is quite fast enough. Most boards run at 300/8/1. This refers to the baud rate, parity and stop bits.

Setting Up

Set the computer up with the modem plugged in and the program running. Dial up the required number and, if the carrier tone is returned, switch the modem to data or on line. The carrier tone which you hear from the distant end is normally a high-pitched whistle. After switching the modem from Telephone to data, it may be necessary to hit the return key a couple of times. This will depend on the type of board to which you are connecting.

Most bulletin boards have a welcome screen, as shown in Figure 1. This will sometimes tell you the hours and the protocols (baud rate, parity, stop bills etc.) needed, but all boards will vary.

At some stage it should come up with FIRST NAME? Understandably, most board operators do not like false or silly names. They have not supplied a lot of expensive equ.pment, time and money for people to abuse the system and take advantage of a free service. Whilst on this note of abuse, it should be noted that many of the earlier boards have closed down or are beginning to charge a subscription fee for their services. The main reason for this is that people are swearing or simply try to crash the program which can cause untold damage to the files and in some cases also the ha-dware. It must be borne in mind that you are a guest to the system and should behave in a responsible manner. It is our hope that communicating by computer will not go the way of many CB channels where appalling language and abuse are the norm.

After entering your first name and htting return – not all boards will require you to press return – you will be asked to enter your last name. The board will then proceed to check its files to see if you are a registered user. Several things may now happen:

a) If you need to be a registered user you may well be cut off.

b) If you are a registered user, it will tell you from where you are calling and request that you enter a password. On receipt of your password it will connect you to the system. It will then check if there are any messages waiting for you.

c) If you are not registered it will probably ask you to fill in a questionnaire ready for you to use on your next call.

As already mentioned, all boards vary, so for now we will concentrate on one in particular.

Terminal Program

The Terminal Program featured in this issue is approximately 14.5K long and there now follows an explanation of the facilities it offers.

It is a completely menu driven program with the option to use tape or disk. The screen colours can be changed to any of those available on the C64 and the border and text colours can also be altered to suit your requirements.

The printer option can be used with either device number four or device number six – the printer plotter.

A 30K buffer for down-loading incoming data from the board is switchable whilst on line and this data can then be saved to tape or disk. A screen full of data, such as the menu, can be saved straight into memory for future reference or likewise be saved to tape or disk. These screens and buffers can then be printed out for a hard copy of your visit to the board.

Since many boards are different, there is also the facility to set up baud file. This file is similar to a mini database i.e. name, telephone number, baud rates and Macro if any. When the baud file which you have selected is loaded in, this will automatically set them up with the required protocols.

Another option available is the ability to pretype a message before going on line, which will drastically cut down time on line and consequently your phone bill. One key will then send out the message at the same time as you receive incoming data. This is very handy when you need to send the same information to a lot of different people.

PROGRAM: COMMS GEN3

```
2000 FOR L=0 TO 176:CX=0
   :FOR D=0 TO 15:READ A
   :CX=CX+A:POKE 5376+L*16+D,
   A:NEXT D
2010 READ A:IF A<>CX THEN PR
   INT"ERROR IN LINE";
   2040+(L*10):STOP
2020 NEXT L
2040 DATA 21,21,0,21,0,21,0,
   21,32,88,79,78,47,88,79,
   70,666
2050 DATA 70,47,0,32,51,76,
   73,78,69,47,0,72,65,76,70,
   47,873
2060 DATA 0,70,85,76,76,47,0,
   32,32,51,48,48,47,0,32,32,
   676
2070 DATA 54,48,48,47,0,32,
   49,50,48,48,47,0,56,47,0,
   55,629
2080 DATA 47,0,54,47,0,53,47,
   0,49,0,50,0,78,79,32,80,
   616
2090 DATA 65,82,73,84,89,0,
   79,68,68,32,80,65,82,73,
   84,89,1113
2100 DATA 0,69,86,69,78,32,
   80,65,82,73,84,89,0,78,79,
   32,996
2110 DATA 80,65,82,73,84,89,
   32,77,0,78,79,32,80,65,82,
   73,1071
2120 DATA 84,89,32,83,0,9,
   255,6,0,0,0,0,0,0,0,0,558
2130 DATA 0,20,0,0,0,0,13,0,
   0,0,0,0,147,0,0,0,180
2140 DATA 0,0,0,0,0,0,0,0,0,
   32,33,34,35,36,37,38,245
2150 DATA 39,40,41,42,43,44,
   45,46,47,48,49,50,51,52,
   53,54,744
2160 DATA 55,56,57,58,59,60,
   61,62,63,64,193,194,195,
   196,197,198,1768
2170 DATA 199,200,201,202,
   203,204,205,206,207,208,
   209,210,211,212,213,214,
   3304
2180 DATA 215,216,217,218,91,
   92,93,94,95,0,65,66,67,68,
   69,70,1736
2190 DATA 71,72,73,74,75,76,
   77,78,79,80,81,82,83,84,
   85,86,1256
2200 DATA 87,88,89,90,91,47,
   93,96,20,0,0,0,1,2,3,4,711
2210 DATA 5,6,7,8,9,10,11,12,
   13,14,15,16,17,18,19,20,
   200
2220 DATA 21,22,23,24,25,26,
   27,28,29,30,31,32,33,34,
   35,36,456
2230 DATA 37,38,39,40,41,42,
   43,44,45,46,47,48,49,50,
   51,52,712
2240 DATA 53,54,55,56,57,58,
   59,60,61,62,63,64,97,98,
   99,100,1096
2250 DATA 101,102,103,104,
   105,106,107,108,109,110,
   111,96,113,114,115,116,
   1720
2260 DATA 117,118,119,120,
   121,122,91,0,93,94,95,0,0,
   0,0,0,1090
2270 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2280 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2290 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2300 DATA 0,0,0,0,0,0,0,0,
   127,0,0,32,0,0,0,0,159
2310 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2320 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,65,66,67,68,266
2330 DATA 69,70,71,72,73,74,
   75,76,77,78,79,80,81,82,
   83,84,1224
2340 DATA 85,86,87,88,89,90,
   91,0,0,0,0,0,0,0,0,0,616
2350 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2360 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,255,255
2370 DATA 32,83,69,84,32,65,
   84,32,0,13,32,32,17,69,78,
   84,806
2380 DATA 69,82,32,84,69,76,
   69,32,78,79,32,32,0,17,17,
   32,800
2390 DATA 32,69,78,84,69,82,
   32,77,65,67,82,79,32,77,
   65,88,1078
2400 DATA 32,51,54,32,67,72,
   82,83,13,0,13,32,32,73,83,
   32,751
2410 DATA 84,72,73,83,32,67,
   79,82,82,69,67,84,63,32,
   89,47,1105
2420 DATA 78,13,17,17,0,147,
   17,32,66,79,65,82,68,83,
   32,78,874
2430 DATA 65,77,69,32,32,32,
   0,0,17,32,32,66,79,65,82,
   68,748
2440 DATA 32,83,69,84,32,65,
   84,13,17,17,0,0,0,0,0,0,
   496
2450 DATA 0,0,0,0,0,0,0,0,0,
   44,49,0,0,0,0,0,93
2460 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2470 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2480 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0
2490 DATA 0,0,0,0,0,32,109,
   65,67,82,79,32,115,69,84,
   32,766
2500 DATA 97,84,45,13,13,32,
   0,64,65,66,67,68,69,70,71,
   72,896
2510 DATA 73,74,75,76,77,78,
   79,80,81,82,83,84,85,86,
   87,88,1288
2520 DATA 89,90,91,92,93,94,
   95,32,33,34,35,36,37,38,
   39,40,968
2530 DATA 41,42,43,44,45,46,
   47,48,49,50,51,52,53,54,
   55,56,776
2540 DATA 57,58,59,60,61,62,
   63,64,97,98,99,100,101,
   102,103,104,1288
2550 DATA 105,106,107,108,
   109,110,111,112,113,114,
   115,116,117,118,119,120,
   1800
2560 DATA 121,122,0,13,1,254,
   170,0,170,0,170,0,170,0,
   170,0,1361
2570 DATA 255,87,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2722
2580 DATA 170,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1360
2590 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2600 DATA 8,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1368
2610 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2620 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2630 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2640 DATA 0,168,0,168,0,168,
```

It is also possible to relive the entire time which you spent on the board at a later stage, without being on line.

Communication with the outside world can be both fun and educational. We hope that you will soon be up and running so you can make full use of your computer.

TELCOM-64 Instructions

To Load TELCOM-64 Type LOAD"*",8,1 or ,1,1 - depending on whether you are using disk or tape. The program will auto-run and you will receive the Main Menu Screen as shown. Hitting RUN/STOP at anytime will take you back to the start of the program. Hitting the back arrow key (top left) will abort reading files and get back to the Main Menu again.

Some explanation may be needed regarding the Functions on the Main Menus so bear with us if you already know.

F1 — Set Protocols (Baud Rate)

On selecting F1 the screen will show a list of options: BAUD RATE/WORD LENGTH/STOP BITS.

Selection required is shown by the > sign next to the words Baud Rate. Select the appropriate speed, i.e. 300, using F1. The > sign will then go to the next selection required - Word Length. After selecting these the Menu will now show Hand shaking, Parity (error checking) and Duplex (both send and receive at the same time). Any combination is

Main Menu	
1. Set Protocols (Baud Rate)	F1
2. Go to Terminal Mode	F2
3. Set up Message Screen	F3
4. View Saved Screens	F4
5. Create a Board File	F5
6. Load/View Board File	F6
7. Load Saved File (view or print)	F7
8. Facility Screen	F8

Selecting Key F8 will take you to another menu, to enable you to set up Device types and Colours etc.

possible. You are then returned to the Main Menu. From here you may choose to go on line by selecting F2.

Facility Menu	
1. Printer Device No. 4 or 6	F1
2. Disk or Tape?	F2
3. Increment Screen Colour	F3
4. Increment Border Colour	F4
5. Increment Cursor Colour	F5
6. Bell ON/OFF	F6
7. Unused on this version	F7
8. Return to Main Menu	F8

Keys F1-5 are self explanatory but a few words may be needed concerning F6 Bell ON/OFF. Some Board Systems will send a control character, which is meant as an audible warning. This may mean that the time left on line is near the limit, or you are nearing the maximum amount of text allowed for a message. A bell will sound if you choose to activate this option.

Key F7 is not used on this version (any ideas for version two?). F8 Returns you to the Main Menu.

F2 — Go to Terminal Mode

The screen clears and you are asked for a file name, under which your screens will be saved e.g. CBBS LONDON. When the directory is listed (disk only) the No: 01 is placed automatically. Every time you save a screen the number will increment by one. To load the screen for later viewing the complete directory name, including the number, must be used. Use F7 to save the

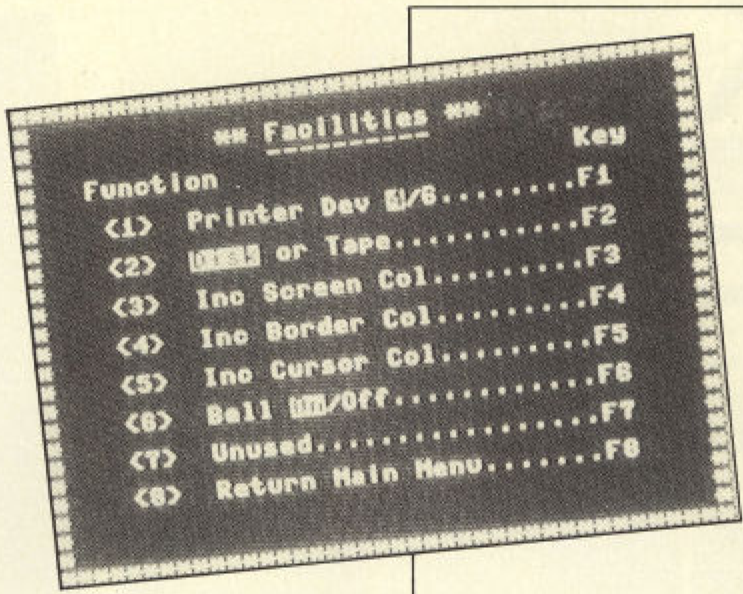
```
   0,170,0,170,0,170,0,170,0,
   170,1354
2650 DATA 85,253,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2718
2660 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2670 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2680 DATA 168,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1358
2690 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2700 DATA 170,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1360
2710 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2720 DATA 168,0,168,0,168,0,
   170,0,168,0,168,0,170,0,
   170,0,1350
2730 DATA 253,85,253,85,255,
   85,255,85,253,85,255,85,
   255,85,255,85,2714
2740 DATA 168,0,168,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1356
2750 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2760 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2770 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2780 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2790 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2800 DATA 0,168,0,168,0,168,
   0,170,0,170,0,170,0,170,0,
   170,1354
2810 DATA 85,253,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2718
2820 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0
   170,1360
2830 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2840 DATA 160,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1350
2850 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2860 DATA 170,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1360
2870 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2880 DATA 168,0,168,0,168,0,
   168,0,168,0,168,0,170,0,
   170,0,1348
2890 DATA 253,85,253,85,255,
   85,255,85,253,85,255,85,
   255,85,255,85,2714
2900 DATA 168,0,168,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1356
2910 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
2920 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2930 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2940 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2950 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
2960 DATA 0,168,0,168,0,168,
   0,170,0,170,0,170,0,170,0,
   170,1354
2970 DATA 85,253,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2718
2980 DATA 0,170,0,170,0,170,
   0,170,0,170,0,170,0,170,0,
   170,1360
2990 DATA 85,255,85,255,85,
   255,85,255,85,255,85,255,
   85,255,85,255,2720
3000 DATA 160,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1350
3010 DATA 255,85,255,85,255,
   85,255,85,255,85,255,85,
   255,85,255,85,2720
3020 DATA 170,0,170,0,170,0,
   170,0,170,0,170,0,170,0,
   170,0,1360
3030 DATA 255,85,255,85,255,
```

```
85,255,85,255,85,255,85,
255,85,255,85,2720
3040 DATA 168,0,168,0,168,0,
168,0,168,0,168,0,170,0,
170,0,1348
3050 DATA 253,85,253,85,255,
85,255,85,253,85,255,85,
255,85,255,85,2714
3060 DATA 168,0,168,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1356
3070 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3080 DATA 2,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1362
3090 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3100 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3110 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3120 DATA 2,170,2,170,2,170,
2,170,2,170,2,170,2,170,0,
170,1374
3130 DATA 87,255,87,255,85,
255,85,255,87,255,85,255,
85,255,85,255,2726
3140 DATA 2,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1362
3150 DATA 87,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2722
3160 DATA 168,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1358
3170 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3180 DATA 170,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1360
3190 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3200 DATA 168,0,168,0,168,0,
168,0,168,0,168,0,170,0,
170,0,1348
3210 DATA 253,85,253,85,255,
85,255,85,253,85,255,85,
255,85,255,85,2714
3220 DATA 168,0,168,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1356
3230 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3240 DATA 2,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1362
3250 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3260 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3270 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3280 DATA 2,170,2,170,2,170,
2,170,2,170,2,170,2,170,0,
170,1374
3290 DATA 87,255,87,255,85,
255,85,255,87,255,85,255,
85,255,85,255,2726
3300 DATA 2,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1362
3310 DATA 87,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2722
3320 DATA 160,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1350
3330 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3340 DATA 170,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1360
3350 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3360 DATA 168,0,168,0,168,0,
168,0,168,0,168,0,170,0,
170,0,1348
3370 DATA 253,85,253,85,255,
85,255,85,253,85,255,85,
255,85,255,85,2714
3380 DATA 168,0,168,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1356
3390 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3400 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3410 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3420 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3430 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3440 DATA 0,168,0,168,0,168,
0,170,0,170,0,170,0,170,0,
170,1354
3450 DATA 85,253,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2718
3460 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3470 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3480 DATA 162,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1352
3490 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3500 DATA 170,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1360
3510 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3520 DATA 170,2,170,2,170,0,
170,0,170,0,170,0,170,0,
170,0,1364
3530 DATA 255,87,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2722
3540 DATA 170,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1360
3550 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3560 DATA 8,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1368
3570 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3580 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3590 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3600 DATA 0,168,0,168,0,170,
0,170,0,170,0,170,0,170,0,
170,1356
3610 DATA 85,253,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2718
3620 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3630 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3640 DATA 168,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1358
3650 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3660 DATA 170,0,170,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1360
3670 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3680 DATA 168,0,168,0,168,0,
170,0,168,0,168,0,170,0,
170,0,1350
3690 DATA 253,85,253,85,255,
85,255,85,253,85,255,85,
255,85,255,85,2714
3700 DATA 168,0,168,0,170,0,
170,0,170,0,170,0,170,0,
170,0,1356
3710 DATA 255,85,255,85,255,
85,255,85,255,85,255,85,
255,85,255,85,2720
3720 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3730 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3740 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3750 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3760 DATA 0,168,0,168,0,168,
0,170,0,170,0,170,0,170,0,
170,1354
3770 DATA 85,253,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2718
3780 DATA 0,170,0,170,0,170,
0,170,0,170,0,170,0,170,0,
170,1360
3790 DATA 85,255,85,255,85,
255,85,255,85,255,85,255,
85,255,85,255,2720
3800 DATA 42,255,255,255,255,
255,255,255,255,255,255,
255,255,255,255,255,3867
4000 PRINT"[CLEAR]NEW"
:PRINT"[DOWN2]
```

screens. On entering the file name, followed by <cr> Return, the screen will now display:

Boards Name ....
Tele No ....
Board Set at.../.../.../../../..
Macro Set at ....

A back arrow (top left key) will now flash as the new cursor. You are now on line, ready to dial the required board. On receiving the carrier tone, switch your Modem from Tele to Data. You should now be up and running (fingers crossed that you put in the right Protocols). Whilst in Terminal mode the Function keys have new functions. Pressing F2 will now bring up another Menu.



Terminal Menu	
1. Save Screen to Memory	F1
2. View Screen (Memory)	F2
3. Toggle Buffer ON	F3
4. Toggle Buffer OFF	F4
5. Send Pre-Typed Message	F5
6. Save File	F6
7. Save a Screen to Disk/Tape	F7
8. Return to Main Menu	F8

F1 will save the entire Screen contents to memory. This could be used as a form of Help Screen whilst on line. For instance when bulletin boards list their functions there could be some differences since not all systems use the same keys. Some Boards use P for pause some use Space Bar etc. Pressing F1 when the menu is displayed will give you a handy reference guide.

F2 will allow you to view the above saved screen whilst on line. If you press F1 again, the screen is lost and replaced by the next screen saved. At this stage it might be as well to use F7. This will save the screen straight to disk or tape so that you can print out at a later date and have a hard copy.

F3 — Toggle Buffer On

This will allow you to allocate a 30K Buffer (block of memory) which will save the entire Data that has come in and gone out whilst on line. This is useful if you wish to relive the whole of your time on line without using your telephone (sorry BT). Using this method, you can print out or save screens of information off line. When F3 is pressed in Terminal mode a bell will sound to let you know that it is now open. Pressing F4 (Toggle Buffer Off) will close the buffer. When the buffer is nearly full, the screen will flash as a warning to close the buffer. You must now save the file to tape or disk.

F5 — Send Pre-Typed Message

By pressing F5 in Terminal mode a Pre-typed Message – done using F3 on the Main Menu – is sent out to line (a lot faster than I can type).

F6 — Save File

This allows you to save a file from a bulletin board to disk or tape. Follow the prompts on screen i.e. File name etc.

F7 — Save Screen

Pressing F7 whilst on line with text coming, enables you to save the screen to disk or tape. Full error checking is given so you will know if any problems occur for instance: disk full, drive not ready or write protect.

```
   LOAD"+CHR$(34)+"COMMS GEN
   4"+CHR$(34)+",8"
4005 REM ** CHANGE ,
   8 IN ABOVE LINE TO ,1 IF
   YOU ARE USING TAPE **
4010 PRINT"[DOWN4]RUN"
4020 POKE 631,13:POKE 632,13
   :POKE 633,13:POKE 198,3
   :PRINT"[HOME]"
```

PROGRAM: COMMS GEN4

```
2000 FOR L=0 TO 80:CX=0
   :FOR D=0 TO 15:READ A
   :CX=CX+A:POKE 8192+L*16+D,
   A:NEXT D
2010 READ A:IF A<>CX THEN PR
   INT"ERROR IN LINE";
   2040+(L*10):STOP
2020 NEXT L
2040 DATA 42,42,42,42,42,42,
   42,42,42,42,42,42,42,42,
   42,42,672
2050 DATA 42,42,42,42,42,42,
   42,42,42,42,42,42,42,42,
   42,42,672
2060 DATA 42,42,42,42,42,42,
   42,42,42,32,32,32,32,32,
   32,32,602
2070 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2080 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2090 DATA 42,62,32,66,1,21,4,
   32,82,1,20,5,19,32,32,32,
   483
2100 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2110 DATA 32,32,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,532
2120 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2130 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2140 DATA 42,32,32,32,32,32,
   32,32,32,32,32,32,32,51,
   48,48,573
2150 DATA 47,51,48,48,46,46,
   46,46,46,46,46,46,46,46,
   46,46,746
2160 DATA 70,49,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,587
2170 DATA 32,32,32,32,32,54,
   48,48,47,54,48,48,46,46,
   46,46,691
2180 DATA 46,46,46,46,46,46,
   46,46,70,51,32,32,32,32,
   32,42,691
2190 DATA 42,32,32,32,32,32,
   32,32,32,32,32,32,49,50,
   48,48,589
2200 DATA 47,49,50,48,48,46,
   46,46,46,46,46,46,46,46,
   46,46,748
2210 DATA 70,53,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,591
2220 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2230 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,42,522
2240 DATA 42,32,32,87,15,18,
   4,32,76,5,14,7,20,8,32,32,
   456
2250 DATA 32,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,32,512
2260 DATA 32,32,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,532
2270 DATA 32,32,32,32,32,32,
   32,56,32,66,9,20,19,46,46,
   46,564
2280 DATA 46,46,46,46,46,46,
   46,46,70,49,32,32,32,32,
   32,42,689
2290 DATA 42,32,32,32,32,32,
   32,32,32,32,32,32,32,32,
   32,55,545
2300 DATA 32,66,9,20,19,46,
   46,46,46,46,46,46,46,46,
   46,46,652
2310 DATA 70,51,32,32,32,32,
   32,42,42,32,32,32,32,32,
   32,32,589
```

BOOK SHELF

Eric Doyle has been scouring the shelves to bring you this look at Commodore books.

Title: Commodore Reference Diary
Author: Jim Butterfield
Publisher: Pitman Publishing
Price: £3.95

THE COMMODORE REFERENCY DIARY by Jim Butterfield may be only pocket sized but it holds something for every Commodore programmer.

Starting with a description of the Commodore range from the venerable PET to the IBM compatible PC and the full range of peripherals, the reference section launches into an overview of the main elements of Basic.

The next section looks at the memory architecture of each machine, giving detailed memory maps of the principal elements and memory locations, including the first breakdown of the C128 which I have seen.

A SuperChart of character, keyword and mnemonic codes in decimal and hex forms a useful conversion chart for quickly translating ASCII codes into screen poke values or for disassembling Basic loaders and many other serious applications.

A whole section is given to several short programs which are useable on any Commodore machine with a few minor alterations, including a simple PET emulator for the C64.

Frequency tables and a few hints and tips on sound generation on the Vic 20 or C64 are then followed by a table of colour codes for these machines and the Plus/4 and C-16. This in turn is followed by a brief look at machine code mnemonics and addressing modes and the section closes with a glossary of computer terms.

As you might expect from the Maharishi of Commodore machines, the diary is aimed at the serious programmer and has already proved to be a handy source of information for me. A fine gift for the programmer who thinks he has everything.

Title: Tool Kit: Kernal
Author: Dan Heeb
Publisher: Compute! Books/Holt Saunders
Price: £13.95

COMPUTE! GAZETTE IS A RATHER EXPensive American magazine which is a mine of information for Commodore, Apple and Atari users, and Compute!'s books of abstracts for each machine are always good value for money.

Tool Kit: Kernal may sound like a collection of utility programs but it is a description of the ROM kernal routines at the end of the memories of the C64 and Vic 20.

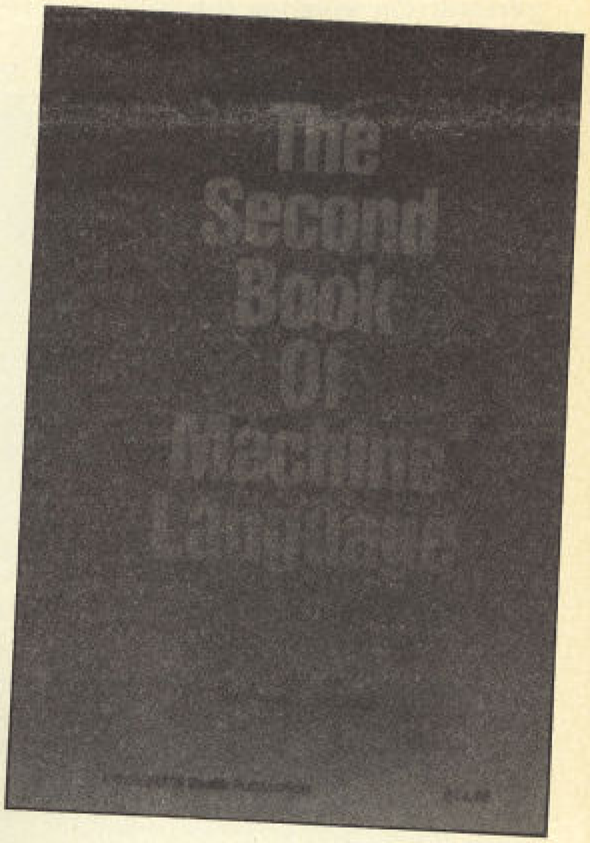


The book is not fully self-contained and Compute!'s Tool Kit: Basic, which deals with the Basic ROM, along with a full disassembly of the kernal memory would be a useful companion to help gain the full benefit from the wealth of information contained in this book.

Each chapter is a grouping of the various subroutines in 'family' groups. For example, all the tape I/O routines form a large chapter which not only describes the routines but explains the way in which a tape file is structured. Other chapters deal in a similar way with interrupts and system reset, screen, serial I/O, RS-232 and principal kernal routines.

Apart from giving details of the subroutines and their relatives, there is a smattering of hints and tips and short useful programs which demonstrate how the kernal may be used within your own projects.

The routines are listed in memory order and abbreviated groupings in two final appendices but a more standard index would have improved the facilities offered by this very useful book.



Title: The Second Book of Machine Language
Author: Richard Mansfield
Publisher: Compute! Books/Holt Saunders
Price: £12.95

IF MACHINE LANGUAGE FOR BEGINners was an introduction to the vocabulary of machine code, then this book is an essay on the grammar of the language. More than this, working through the book gradually builds up a powerful assembler program which not only teaches the basics of machine code programming structure but also acts as a useful addition to your utility library for creating your own routines and programs.

As the name suggests, the Label Assembler Development System (LADS) allows the use of labels and comments within a program which looks like a normal Basic program except that it uses its own mnemonic language based on normal machine code mnemonics.

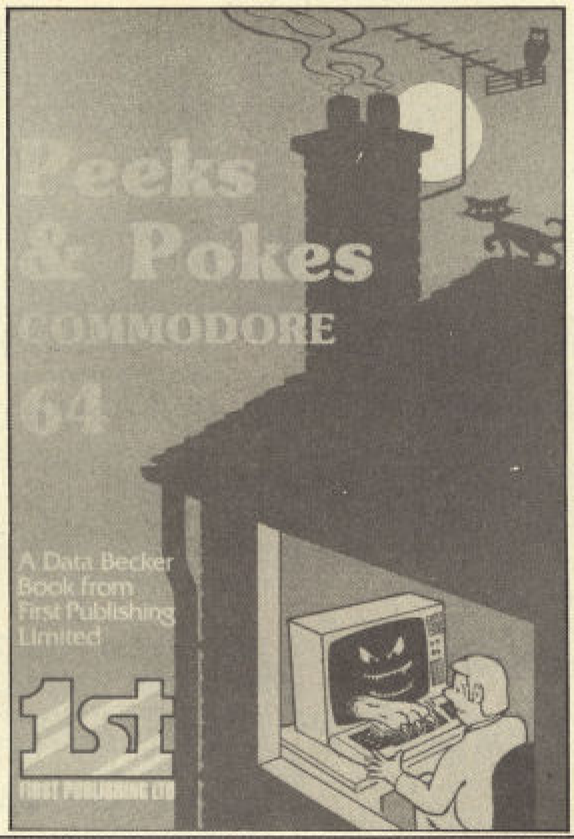
This forms a much easier way to construct a machine code program and not being in code itself means that it can be used to show many of the facets of programming including communicating with peripheral devices and maintaining a database.

The book is written in intelligent English which assumes nothing more than a bit of common sense and perseverance on the part of the reader. Perseverance comes in useful because you have to enter several pages of data which lacks the customary checksum features of other books in the Compute! series.

An appendix of useful subroutines for incrementing, adding and subtracting double byte numbers, multi-byte addition and multiplication and division

adequately cover areas which can be problematic to beginners.

Undoubtedly, the contents of this book offers great value for money to anyone interested in a good introduction to machine language.



Title: Peeks and Pokes Commodore 64
Author: Data Becker Books
Publisher: First Publishing
Price: £7.95

THE MOST DIFFICULT WORDS TO understand in the Basic language must be PEEK and POKE which is a pity, because programming the 64 to any degree of competence soon necessitates their use. Stripping away some of the mystique surroundings these words is the aim of this book from First Publishing.

Starting off with the general basics of computer memory it soon leaps into specific examples on the kind of information which can be obtained by peeking around the machine and how pokes can be used to great effect. The whole of the RAM operating system is scrutinised as the chapters progress with sections on zero page locations, sound and graphics.

Realising that peeks and pokes are the basis of machine code, allows the author to give a gentle introduction to machine language, Basic extensions and games routines. I must stress however that this is just an introduction, not an in-depth study, of these subjects for those who wish to understand the principles without getting too involved in detail.

In some ways I think that the book tries to cover too much in too short a space and the clear explanations which typify the opening chapters give way to a more fuzzy style in the latter part of the book. A painfully slow machine code simulator is included which covers an area better suited to a good machine code monitor with trace facilities.

Despite these criticisms, the book has a lot to offer an inexperienced programmer looking for new ideas.

BOOK SHELF

TOP DR

Allen Webb with more from the fascinating world of sprites.

YOU CAN MOAN AS MUCH as you like about the poor Basic of the C64 but one area which gives it great advantages over many other machines is the provision of sprites. These objects are the life blood of many games authors (just watch one of Mr Minter's games) and can make life very simple. The draw back, as ever, is that a fairly large number of POKEs are necessary to get them to do anything. In this article I will provide a set of routines which will enable you to manipulate sprites with ease.

You will notice that I haven't provided a command for the design of sprites. The reason is that such commands are ridiculously wasteful on memory. If you want to include sprite designs in your program. I suggest the use of DATA statements or a straight block of data. Most of the decent sprite designer programs around enable you to do one or the other (some offer both).

As always, the code is loaded from a Basic loader (Loader 1) and sits in the spare area between ROMs starting at $C000. The commands are called from a jump table at the start of the code. In summary, the calls are:

1. Specify Sprite

SYS SA, SN, Type, Colour, Xexp, Yexp, priority, [colour1, colour2]

Where:
SN is the sprite number (0-7).
Type is the graphics mode 0=high resolution, 1=multi-colour.
Colour is the sprite colour
Xexp, Yexp are flags for sprite expansion, 0=normal, 1= expanded.
Priority is 0 for sprites behind text and 1 for sprites in front.
Colour1 and colour2 are the other colours, only required if type=1.
This command sets up the main parameters for any sprite.

2. On/Off

SYS SA+3,SN,Flag

where SN is the sprite involved. Flag is 1 to turn on a sprite and 0 to turn it off.

3. Position Sprite

SYS SA+6,SN,X,Y

This commands puts a sprite SN at co-ordinates X and Y.

4. Pattern

SYS SA+9,SN,Slot Number

This command sets the sprite pointer for sprite SN to the pattern slot specified.

5. Erase

SYS SA+12,Slot Number

This erases the specified pattern slot.

6. Fill

SYS SA+15,Slot Number, Byte value

This fills the pattern with the specified byte value.

7. Reverse

SYS SA+18,Slot Number

This reverses the specified pattern so that dots become spaces and spaces become dots.

8. Copy

SYS SA+21,Slot Number, pattern address, flag

This copies a pattern of 63 bytes, starting at the specified address to the specified pattern slot. The flag determines how the pattern and the contents of the slot are combined:

0 – overwrites the contents of the pattern slot
1 – EORs the pattern slot
2 – ORs the pattern slot
3 – ANDs the pattern slot

9. Roll Left

SYS SA+24,Slot Number, number of shifts

This moves the specified pattern to the left and the pattern reappears on the right to give a roll. The number of shifts decides how much is scrolled and allows the scrolling of multicolour sprites without odd colour effects (by using an even number of shifts).

10. Roll Right

SYS SA+27, Slot Number, number of shifts

11. Randomise

SYS SA+30,Slot Number

Fills the specified pattern slot with a random pattern.

12. Invert

SYS SA+33,Slot Number

Turns the specified pattern slot upside down.

13. Sprite/Sprite Collision

SYS SA+36,SN

Checks the collision status of sprite SN and returns a value in location 1000. A zero value means no collision, a non zero value means a collision.

14. Sprite/Background Collision

SYS SA+39,SN

Acts in the same way as the sprite/sprite collision but uses location 1001.

15. Animate

SYS SA+,Slot Number, Start Address, Number of frames, delay1,delay2

This displays a sequence of designs in the specified pattern slot. The start address of the patterns is specified and the patterns mus follow as a sequence of 64 bytes. The delay values determine the animation speed. The approximate times to display eight frames are:

D1*D2	Time to display 8 frames (jiffies)
2500	10
5000	15
10000	29
40000	109
62500	168

This demonstration shows the use of the animate command to show a man walking across the room. When he hits the wall, the explosion effect uses the randomise and scroll commands. Don't forget that before you use the collision detect commands, you must clear the collision registers at the start of the program (Line 120 of the demonstration does this). If you omit this step, you will get spurious collision effects.

Most of you will be content to use the normal set of sprites. In fact this set allows a reasonably long Basic program. If you want a longer program, you have two options:

1) Use an alternative memory bank and screen position. The machine code automatically takes the screen position into account when setting the sprite pointers.

2) Simply store your sprite patterns above the Basic memory with the top of Basic dropped to protect them. You can then use the COPY command to extract the patterns.

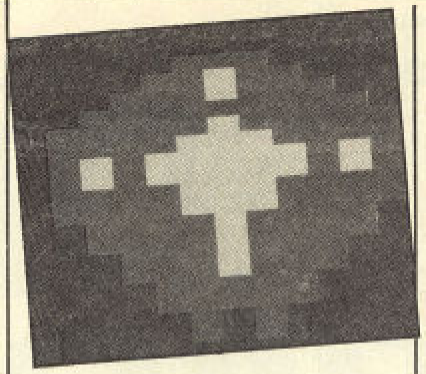
Option one is probably best if you want to use eight sprites on screen at once.

I suggest you look at the Programmer's Reference Guide for information on how to change banks and screen position. Given the energy and time, I'll try to cover this matter in a later article.

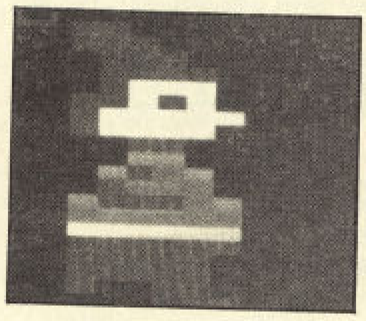
Well, that's it for this month. Next month, perhaps a little snippet on 3D graphics for you lovers of mazes and dungeons and dragons.

PROGRAM: SPRITE LOADER

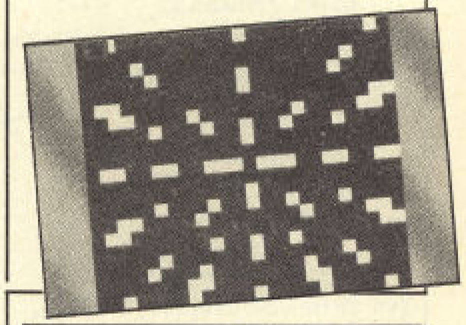
```
1 DATA 76,68,195,76,45,192,
   76,99,192,76,18,193,76,61,
   193,76,80,193,76,98,193
2 DATA 76,115,193,76,108,195,
   76,141,195,76,93,194,76,
   160,194,76,232,194,76
3 DATA 248,194,76,174,195,32,
   18,195,32,8,195,165,20,41,
   1,240,13,172,133,3,185
4 DATA 83,192,13,21,208,141,
   21,208,96,172,133,3,185,
   91,192,45,21,208,141,21
5 DATA 208,96,1,2,4,8,16,32,
   64,128,254,253,251,247,
   239,223,191,127,32,18,195
6 DATA 32,8,195,165,20,164,
   21,141,134,3,140,135,3,32,
   8,195,165,20,141,136,3
7 DATA 173,133,3,24,10,168,
   173,134,3,153,0,208,173,
   136,3,200,153,0,208,173
8 DATA 135,3,240,13,172,133,
   3,185,83,192,13,16,208,
   141,16,208,96,172,133,3
9 DATA 185,91,192,45,16,208,
   141,16,208,96,41,1,240,13,
   172,133,3,185,83,192
10 DATA 13,28,208,141,28,208,
   96,172,133,3,185,91,192,
   45,28,208,141,28,208,96
11 DATA 32,8,195,165,20,41,1,
   240,15,172,133,3,185,83,
   192,13,29,208,141,29,208
12 DATA 76,239,192,172,133,3,
   185,91,192,45,29,208,141,
   29,208,32,8,195,165,20
13 DATA 41,1,240,13,172,133,
```



```
   3,185,83,192,13,23,208,
   141,23,208,96,172,133,3
14 DATA 185,91,192,45,23,208,
   141,23,208,96,32,18,195,
   32,8,195,165,20,133,251
15 DATA 173,136,2,24,105,4,
   133,254,169,0,133,253,56,
   165,253,233,8,133,253,165
16 DATA 254,233,0,133,254,
   172,133,3,165,251,145,253,
   96,32,33,195,169,0,160
17 DATA 0,145,253,200,192,64,
   208,249,96,192,128,64,0,
   32,33,195,32,8,195,165
18 DATA 20,160,0,145,253,200,
   192,64,208,249,96,32,33,
   195,160,0,177,253,73,255
19 DATA 145,253,200,192,64,
   208,245,96,32,33,195,32,8,
   195,165,20,164,21,133
20 DATA 251,132,252,32,8,195,
   166,20,160,0,165,1,41,254,
   133,1,177,251,224,0
21 DATA 240,20,224,1,240,9,
   224,2,240,10,49,253,76,
   168,193,81,253,76,168,193
22 DATA 17,253,145,253,200,
```



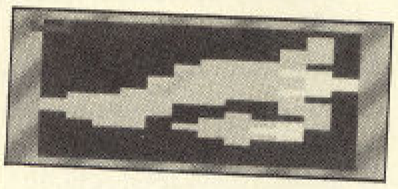
```
   192,64,208,223,165,1,9,1,
   133,1,96,32,38,195,162
23 DATA 0,160,2,177,253,153,
   142,3,136,16,248,14,144,3,
   46,143,3,46,142,3,144
24 DATA 8,169,1,13,144,3,141,
   144,3,160,2,185,142,3,145,
   253,136,16,248,232,224
25 DATA 21,240,16,24,165,253,
   105,3,133,253,165,254,105,
   0,133,254,76,187,193
26 DATA 96,32,38,195,162,0,
   160,2,177,253,153,142,3,
   136,16,248,78,142,3,110
27 DATA 143,3,110,144,3,144,
   8,169,128,13,142,3,141,
   142,3,160,2,185,142,3,145
28 DATA 253,136,16,248,232,
   224,21,240,16,24,165,253,
   105,3,133,253,165,254,105
29 DATA 0,133,254,76,253,193,
   96,32,8,195,165,20,41,1,
   240,13,172,133,3,185,83
30 DATA 192,13,27,208,141,27,
   208,96,172,133,3,185,91,
   192,45,27,208,141,27,208
31 DATA 96,32,33,195,169,0,
   141,142,3,32,121,194,172,
   142,3,145,253,200,192,64
32 DATA 240,6,140,142,3,76,
   101,194,96,169,255,141,14,
   212,141,15,212,169,128
33 DATA 141,18,212,169,128,
   141,24,212,173,27,212,96,
   32,8,195,165,20,141,37
34 DATA 208,32,8,195,165,20,
   141,38,208,96,32,33,195,
   169,29,160,35,141,139,3
35 DATA 140,140,3,169,3,141,
   137,3,172,139,3,177,253,
   141,138,3,172,140,3,177
36 DATA 253,172,139,3,145,
   253,173,138,3,172,140,3,
   145,253,206,139,3,206,140
37 DATA 3,206,137,3,208,219,
   173,139,3,48,11,173,140,3
   24,105,6,141,140,3,208
38 DATA 198,96,32,18,195,172
   133,3,185,83,192,45,30,
   208,141,232,3,96,32,18
39 DATA 195,172,133,3,185,83
   192,45,31,208,141,233,3,
   96,32,253,174,32,138,173
40 DATA 32,247,183,96,32,8,
   195,165,20,201,8,144,2,
   169,7,141,133,3,96,32,8,
   195
41 DATA 165,20,133,253,169,0,
   133,254,162,6,6,253,38,
   254,202,208,249,173,0,221
42 DATA 41,3,168,185,76,193,
   24,101,254,133,254,96,32,
   18,195,32,8,195,165,20
43 DATA 141,141,3,32,173,192,
   32,8,195,165,20,172,133,3,
   153,39,208,32,203,192
44 DATA 32,58,194,173,141,3,
   240,3,32,143,194,96,32,8,
   195,165,20,141,182,3,32
45 DATA 8,195,165,20,141,183,
   3,240,14,173,182,3,32,182,
   193,206,183,3,173,183
46 DATA 3,208,242,96,32,8,
   195,165,20,141,182,3,32,8,
   195,165,20,141,183,3,240
47 DATA 237,173,182,3,32,248,
   193,206,183,3,173,183,3,
   208,242,96,32,33,195,32
48 DATA 8,195,165,20,164,21,
   133,251,132,252,32,8,195,
   165,20,141,138,3,32,8
49 DATA 195,165,20,141,194,3,
   32,8,195,165,20,141,195,3,
   32,242,195,206,138,3
50 DATA 173,138,3,201,0,240,
   16,24,165,251,105,64,133,
   251,165,252,105,0,133
51 DATA 252,76,212,195,96,
   160,0,165,1,41,254,133,1,
   177,251,145,253,200,192
52 DATA 64,208,247,165,1,9,1,
   133,1,174,194,3,172,195,3,
   136,208,253,202,208
53 DATA 247,96,20
54 FOR I=49152 TO 50198
55 READ X: POKE I,X: T=T+X
56 NEXT
57 IF T<>122766 THEN PRINT"E
   RROR IN DATA"
58 REM ****************
59 REM *[SPC2]SPRITE AID
   [SPC2]*
60 REM *[SPC3]AEW 1985[SPC3]*
61 REM ****************
```



PROGRAM: SPRITE DEMO

```
10 REM **********************
   **********
20 REM * DEMONSTRATION OF
    SPRITE AID *
30 REM **********************
   **********
40 GOSUB 1250 : REM LOAD SPR
   ITES STARTING AT 12288
50 PRINT"[CLEAR,RIGHT29,
   DOWN2,RVSON,SPC,DOWN,LEFT,
   SPC,DOWN,LEFT,SPC,DOWN,
   LEFT,SPC,DOWN,LEFT,SPC,
   DOWN,LEFT,SPC,DOWN,LEFT,
   SPC,DOWN,LEFT,SPC,DOWN,
   LEFT,SPC,DOWN,LEFT,SPC]"
60 PRINT"[C+31]"
70 SA=12*4096:POKE 1001,0
```

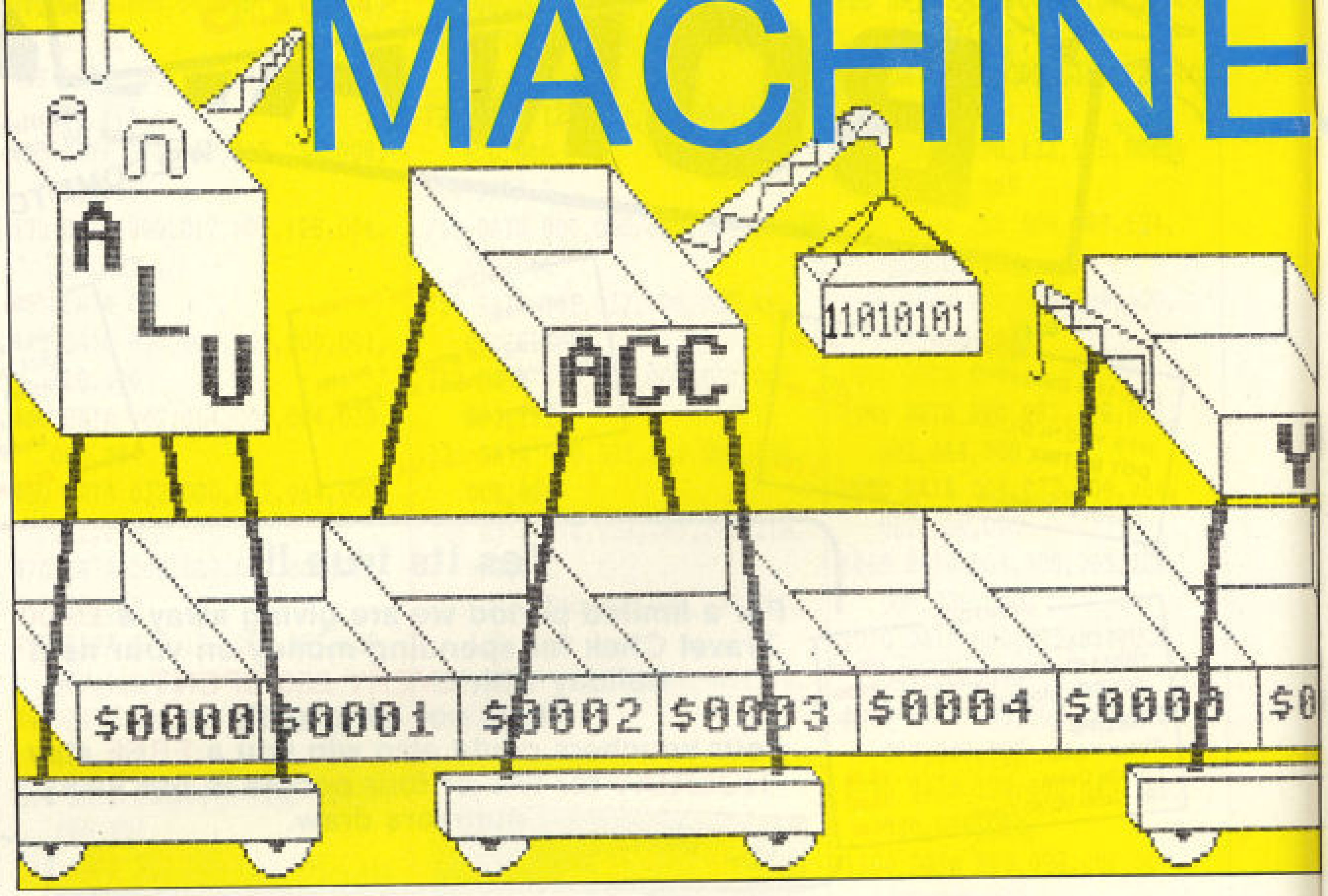
```
80 SYS SA,1,0,1,1,1,0
   : REM SET UP SPRITE 1
90 SYS SA+3,1,1: REM TURN
   SPRITE 1 ON
100 SYS SA+6,1,40,104
   : REM PUT IT AT 40,104
110 SYS SA+9,1,13
   : REM SPRITE 1 TO PATTERN
   13
120 X=40:A=PEEK(53279)
130 SYS SA+42,13,12288,4,150,
   120: REM RUN THROUGH FIRS
   T FOUR FRAMES
140 X=X+4:SYS SA+6,1,X,104
   : REM[SPC2]MOVE ALONG A
   BIT
150 SYS SA+42,13,12608,4,150,
   120: REM THE NEXT FOUR
   FRAMES
160 X=X+4:SYS SA+6,1,X,104
170 SYS SA+39,1:IF PEEK(1001
   )>0 THEN 190: REM HAVE
   WE HIT THE WALL?
180 GOTO 130: REM NO - CARRY
   ON
190 SYS SA,1,1,7,1,1,0,2,1
   : REM CHANGE SPRITE TO
   MULTICOLOUR
200 FOR J=1 TO 100
   :SYS SA+30,13:NEXT
   : REM EXPLOSION EFFECT
   USING RANDOMISE
210 FOR J=1 TO 100
   :SYS SA+24,13,1:NEXT
   :SYS SA+12,13
   : REM SCROLL AND TURN OFF
   SPRITE
220 END
230 REM SPRITES[SPC9]
240 DATA 000,000,000,000,003,
   000,000
250 DATA 004,128,000,008,064,
   000,008
260 DATA 064,000,004,128,000,
   007,000
270 DATA 000,122,000,001,132,
   000,002
280 DATA 114,000,004,162,000,
   009,067
290 DATA 096,006,031,224,000,
   133,128
300 DATA 000,132,000,001,136,
   000,031
310 DATA 228,000,063,146,192,
   028,019
320 DATA 064,012,012,064,004,
   003,128
330 DATA 0
340 DATA 000,000,000,000,000,
   000,000
350 DATA 001,128,000,002,064,
   000,004
360 DATA 032,000,004,032,000,
   002,064
370 DATA 000,005,128,000,058,
   000,000
380 DATA 193,000,001,025,000,
   002,025
390 DATA 000,005,067,096,002,
   131,224
400 DATA 000,130,192,000,132,
   000,013
410 DATA 210,000,015,233,000,
   015,006
420 DATA 000,012,009,128,006,
   007,192
430 DATA 0
440 DATA 000,000,000,000,001,
   128,000
450 DATA 002,064,000,004,032,
   000,004
460 DATA 032,000,002,064,000,
   001,128
470 DATA 000,007,000,000,024,
   128,000
480 DATA 034,128,000,092,128,
   000,161
490 DATA 000,000,081,064,000,
   115,192
500 DATA 000,069,128,000,052,
   000,003
510 DATA 236,000,003,202,000,
   003,020
520 DATA 000,001,171,000,000,
   061,000
530 DATA 0
540 DATA 000,001,128,000,002,
   064,000
550 DATA 004,032,000,004,032,
   000,002
560 DATA 064,000,005,128,000,
   006,000
570 DATA 000,025,000,000,037,
   000,000
580 DATA 073,000,000,008,000,
   000,082
590 DATA 000,000,043,000,000,
   053,000
600 DATA 000,076,000,000,046,
   000,000
610 DATA 236,000,000,072,000,
   000,048
620 DATA 000,000,078,000,000,
   124,000
630 DATA 0
640 DATA 000,000,000,000,001,
   128,000
650 DATA 002,064,000,004,032,
   000,004
660 DATA 032,000,002,064,000,
   005,128
670 DATA 000,026,000,000,061,
   000,000
680 DATA 245,000,001,246,000,
   001,229
690 DATA 192,000,194,032,000,
   069,224
700 DATA 000,070,000,000,047,
   000,000
710 DATA 043,000,000,087,192,
   000,083
720 DATA 128,000,140,000,000,
   120,000
730 DATA 0
740 DATA 000,000,000,000,000,
   192,000
750 DATA 001,032,000,002,016,
   000,002
760 DATA 016,000,001,032,000,
   002,192
770 DATA 000,003,000,000,029,
   000,000
780 DATA 248,128,001,242,216,
   003,162
790 DATA 040,003,065,208,000,
   129,000
800 DATA 000,130,000,000,076,
   000,013
```



```
810 DATA 188,000,010,078,064,
   009,007
820 DATA 192,004,131,192,003,
   192,128
830 DATA 0
840 DATA 000,000,000,000,000,
   000,000
850 DATA 000,192,000,001,032,
   000,002
860 DATA 016,000,002,016,000,
   001,032
870 DATA 000,002,192,000,005,
   000,000
880 DATA 058,128,000,250,128,
   001,242
890 DATA 088,003,065,040,003,
   064,232
900 DATA 000,129,016,000,130,
   000,027
910 DATA 031,000,020,239,128,
   019,007
920 DATA 000,010,015,192,006,
   007,128
930 DATA 0
940 DATA 000,000,000,000,003,
   000,000
950 DATA 004,128,000,008,064,
   000,008
960 DATA 064,000,004,128,000,
   011,000
970 DATA 000,010,000,000,026,
   000,000
980 DATA 114,000,001,229,000,
   001,073
990 DATA 000,000,136,192,000,
   134,064
1000 DATA 000,133,128,006,
   088,000,009
1010 DATA 152,000,009,124,
   000,010,152
1020 DATA 000,004,064,000,
   000,060,000
1030 DATA 0
1040 DATA 000,001,128,000,
   002,064,000
1050 DATA 004,032,000,004,
   032,000,002
1060 DATA 064,000,005,128,
   000,006,000
1070 DATA 000,025,000,000,
   049,000,000
1080 DATA 105,000,000,106,
   000,000,107
1090 DATA 128,000,040,128,
   000,071,000
1100 DATA 000,098,000,000,
   025,000,000
1110 DATA 105,000,000,082,
   000,000,076
1120 DATA 000,000,040,000,
   000,030,000
1130 DATA 0
1140 DATA 000,000,000,000,
   001,128,000
1150 DATA 002,064,000,004,
   032,000,004
1160 DATA 032,000,002,064,
   000,005,128
1170 DATA 000,026,000,000,
   033,000,000
1180 DATA 201,000,001,026,
   000,001,035
1190 DATA 192,000,163,192,
   000,069,128
1200 DATA 000,066,000,000,
   057,000,000
1210 DATA 053,128,000,116,
   064,000,115
1220 DATA 128,000,248,000,
   000,124,000
1230 DATA 0
1250 FOR I=12288 TO 12927
1260 READ X:POKE I,X:NEXT
1270 RETURN
```

More on the mysteries of machine code from Allen Webb.

WELCOME TO THE MACHINE



SO, WHAT EXACTLY IS machine code? Well it's a set of instructions which the microprocessor can carry out. These instructions are stored as eight bit numbers, so that a maximum of 256 instructions are available. In reality, the number of instructions is significantly less than 256.

Each instruction, or op-code, may be followed by an eight or 16 bit data value (operand) depending on the type and mode of the instruction. Each complete instruction will therefore occupy one, two or three bytes.

We humans are basically stupid creatures and it's obviously not feasible to write machine code directly as a sequence of numbers. Each instruction is therefore represented by a mnemonic which aids programming. Some examples of mnemonics are:

LDA – LoaD Accumulator
TAX – Transfer Accumulator to X register

The example given last month shows the use of mnemonics. We can now outline the programming steps involved in writing machine code:

1. Using an editor, write a program using mnemonics and labels to ease the process. This is called the SOURCE CODE.
2. Convert the source code to the numerical instructions (OBJECT CODE) which are used by the processor. This conversion is performed by the assembler.
3. Load, debug and use the object code.

That's all there is to it. Many people try to attach a lot of mystery and difficulty to machine code but provided you use a disciplined approach, machine code programming is not complicated.

Before we start work, it's time to adopt some new attitudes to programming. First, forget all the bad habits that you may have learnt with Basic. Basic may tolerate sloppy programming, machine code won't. If you fill the stack, for example, Basic will give you an OUT OF MEMORY error – machine code will crash.

Here are some tips:

1. Before you start coding, prepare a detailed program algorithm or flowsheet on paper.
2. If the program is large, split it up into small, easily debugged modules.
3. Use copious amounts of remarks to tell you the function of each section of the program and keep a track of what actions each module performs and which registers/memory locations are affected.
4. Save your source code regularly to prevent the loss of hours of work by a crash.
5. Keep smiling!

Compared to high level languages, machine code instructions are rather primitive and quite a number are required to perform simple tasks. This does explain why Basic is so sluggish but it is also what makes machine code versatile and often quite compact.

The instructions mainly involve the movement and manipulation of data, with the various registers operating in a simple hierarchy as implied in Figure 1.

1. The ALU is the big boy of the system being able to perform actions on both RAM and the other registers.
2. Next down the heap comes the accumulator which is able to act on RAM and the X and Y registers.
3. The X and Y registers can transfer data from RAM or the accumulator but are limited in power.
4. Finally, RAM has no capabilities and simply acts as a series of boxes for the storage of data.

All data transfers are carried out via the accumulator or X and Y registers. The type of data transfer depends on the "addressing mode". The 6502/6510 have 13 addressing modes making them quite versatile.

OK, let's make a start on the commands for loading and saving the eight bit internal registers.

The simplest addressing mode is immediate mode. This mode involves the loading of a specified data value into a register and is signified by prefixing the data value b #. For example:

LDA #20

This insertion loads the accumulator with 20. The X and Y registers can also be used in immediate mode:

LDA # $A0 (loads the X register with hex $A0 (160))
LDY #%00000001 (loads the Y register with binary 1)

A slightly more advanced mode allows the manipulation of the contents of a specified memory location. This is called absolute mode. If the operande is not prefixed by #, this mode is assumed. Here are some examples:

LDA 1024 – loads the accumulator with the contents of location 1024.
LDY $8000 – loads the Y register

with the contents of location $8000.

The contents of a register may also be stored to a memory location in absolute mode.

STA $0400 – stores the accumulator in location $0400.
STX 1000 – stores the X register in location 1000.

Consider this rather contrived example:

```
20 C000 *=$C000
30 C000 A9 01 LDA # 1
40 C002 8D 00 04 STA $0400
50 C005 AC 00 04 LDY $0400
60 C008 8C 00 D8 STY $D800
70 C00B 60 RTS
```

This example does the following:
LINE 10 – Loads the accumulator with the value 1. This uses immediate mode.
LINE 20 – Stores the accumulator in location 1024 (i.e. absolute mode). In other words, location 1024 now holds 1.
LINE 30 – Loads the Y register with the contents of 1024, i.e. 1.
LINE 40 – Stores the Y register in location $D800.
This program is equivalent to:

POKE 1024,1: POKE 55296,1

but is much faster.

Before you experts out there put in the boot, this example routine is deliberately inefficient to simply show the use of immediate and absolute addressing modes.

Let us have a quick look at line 40. The machine code and mnemonics are:

8D 00 04 STA $0400

The first hexadecimal byte, $8D, is the code for the instruction Store Accumulator in absolute mode. The second two bytes are the address of the location involved but reversed i.e. 00 04 is $0400 backwards. This is a convention adopted by this particular processor and is something which we will meet again.

The instruction RTS in line 70 stands for ReTurn from Subroutine. This is similar to the Basic word RETURN.

The processor considers the memory to comprise of a series of 256 byte "pages". The most important page is the block of memory from $0000 to $00FF. This is called the "zero" page and it has its own set of addressing modes. Any decent assembler will check the operand for a value in the zero page and assign the appropriate op-code value. Apart from the size of the operand, zero page address mode is written in the same way as normal absolute mode E.G.

AD 00 04 LDA $0400 – non zero page
A5 40 LDA $0040 – zero page

The value of zero page addressing is that it is faster. A normal absolute accumulator load for example takes four microsounds whilst a zero page load three microseconds on a one megahertz computer.

Table 1

	Immediate	Absolute	Zero Page	Implied
LDA	*	*	*	*
STA	*	*	*	*
LDX	*	*	*	*
STX	*	*	*	*
LDY	*	*	*	*
STY	*	*	*	*
TAX				*
TXA				*
TAY				*
TYA				*
INY				*
DEY				*
INX				*
DEX				*

The importance of the zero page is reflected in its extensive use by the BASIC operating system. The zero page also features strongly in the more cunning addressing modes.

The last of the simple data transfer commands involves the accumulator and the two registers. These are quite self explanatory:
TAX – transfer the contents of the accumulator to the X register.
TAY – transfer the contents of the accumulator to the Y register.
TXA – transfer the contents of the X register to the accumulator.
TYA – transfer the contents of the Y register to the accumulator.

In fact these instructions perform a copying action and both registers involved will contain the same contents. For example, assume that you want to load both the accumulator and the Y register with 255. The following sequence will do the job:

```
LDA # $FF
TAY
```

Alternatively, you could use:

```
LDA # $FF
LDY  $FF or:
LDY  $FF
TYA
```

There is little to choose between the three examples except that the first and last occupy only three bytes (compared to the second which uses four bytes). The second example is also slightly slower.

These instructions use "Implied Mode", as does RTS described earlier. Four other very useful Implied Mode instructions are available to change the values of the X and Y registers. These are used in connection with the registers functions as counters or indices. They increment and decrement the registers by 1:

INY – Increment Y register by 1
DEY – Decrement Y register by 1

Similarly, INX and DEX act on the X register.

As you will see later, the four modes described today are also used by other instruction types.

That's enough for now, table 1 summarises the modes covered this time and which instructions use them.

Starting next month, we will be using larger programs and I will use the Hypabasic assembler given recently in Your Commodore. That way, you will be able to try out the examples.

Your homework this time shouldn't be too much of a problem.

1. What does this routine do?

```
LDA # 3
STA $0400
LDA # 1
STA $0401
LDA # 20
STA $0402
LDA # 1
STA $D800
STA $D801
STA $D802
RTS
```

2. Write a routine which will change the screen to the colour specified in any RAM location (e.g. 1000).
3. Write a routine to rapidly flash the screen with the colour sequence RED, CYAN and PURPLE.

And now, here are the answers to the homework from last month.

1. You should have no problems here:

Decimal	**Binary**
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010

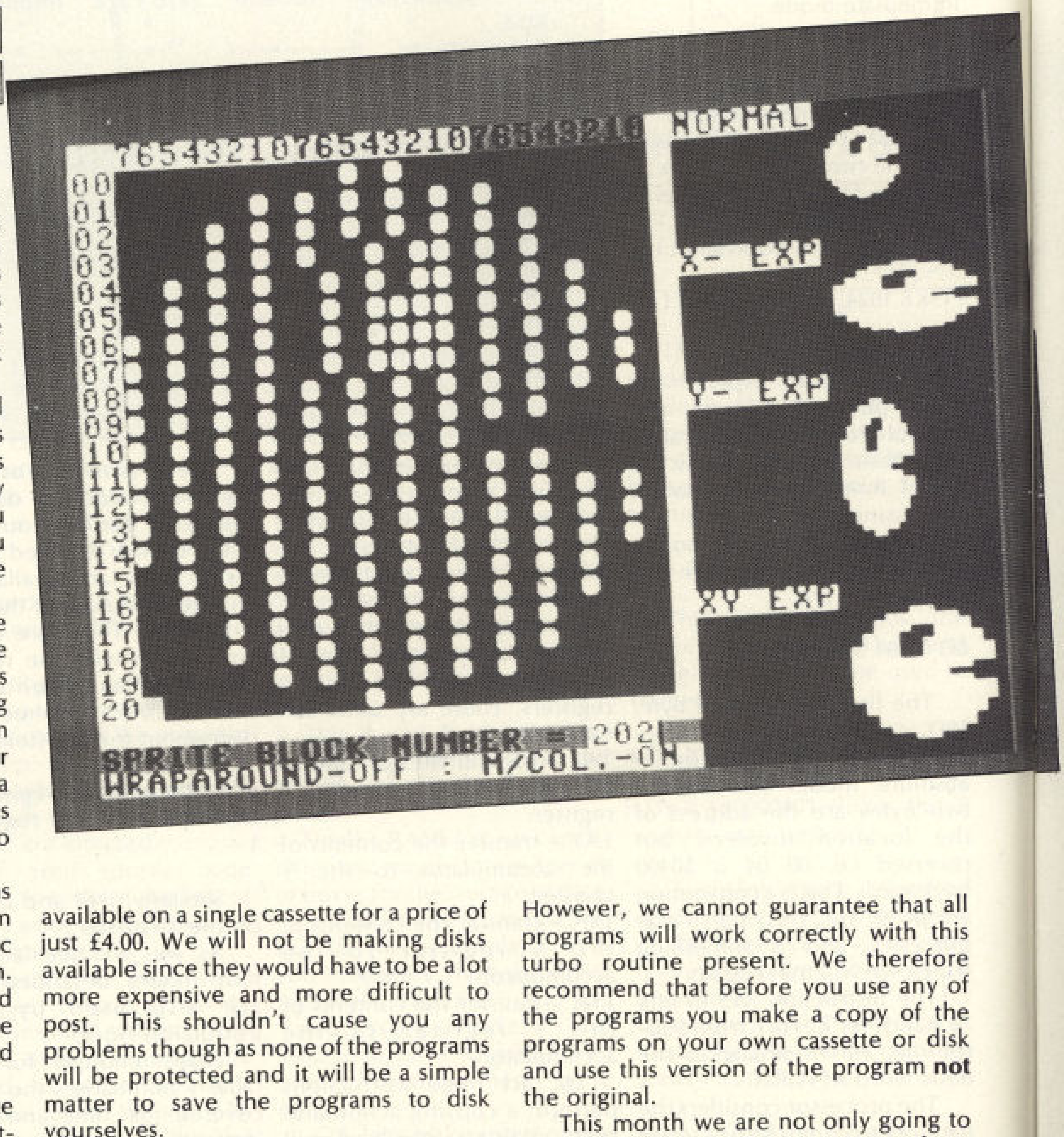
2. %10101010 = 170
123 = 01111011
%111101111 = 495
$4100 = 16640
1123 = $0463
3. As with any program, there is an infinite variety of answers. This example accepts a binary number with or without a % prefix:

```
10 INPUT "BINARY NUMBER";
BN$: BL=LEN(BN$)
20 IF LEFT$(BN$,1)="%" THEN
BL=BL-1: BN$=RIGHT$(BN$,BL)
30 DN=0: PRINT CHR$(147):
FOR BI=0 to BL-1
40 BI$=MID$(BN$,BL-BI,1): IF
BI$<"0"ORBI$>"1" THEN
PRINT "ONLY 1 OR 0
ALLOWED": END
50 DN=DN+VAL(BI$)@2 BI
60 NEXT
70PRINT "%"BN$ ="DN
```

Next month we'll move on to the more complex addressing modes and by using branching we'll write some text and learn some other tricks.

SOFTWARE FOR SALE

Save your fingers a lot of work with our new software service.



IT'S THREE O'CLOCK IN THE MORNING. You sit at the computer keyboard having just finished a marathon typing session entering one of the superb programs from Your Commodore. Your fingers reach for the keyboard and press the letters R, U and N/. You sit back expectantly and...nothing happens.

Well, I'm sure that we have all had problem before now. When it does happen it's a matter of spending hours searching through the program for any typing mistakes. No matter how long you look or how many people help you, you can usually guarantee that at least one little bug slips through unnoticed.

Here, at Your Commodore, we pride ourselves on the quality of listing that we print. Unfortunately, this usually means that they are also very long, thus taking longer to type in and leaving more room for errors. All of the listings in Your Commodore are taken straight from a printout of working programs, it is therefore very unusual for errors to appear in the magazine.

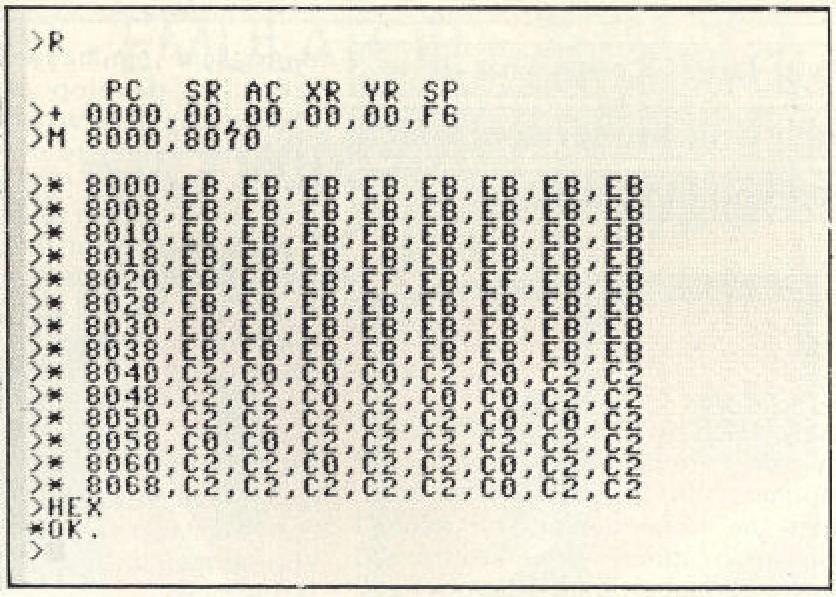
Because of the length of our programs we do get a large number of requests from readers who would like us to put specific programs on tape or disk for them. Obviously this is very time consuming and means that we can't spend as much time working on the magazine as we would like.

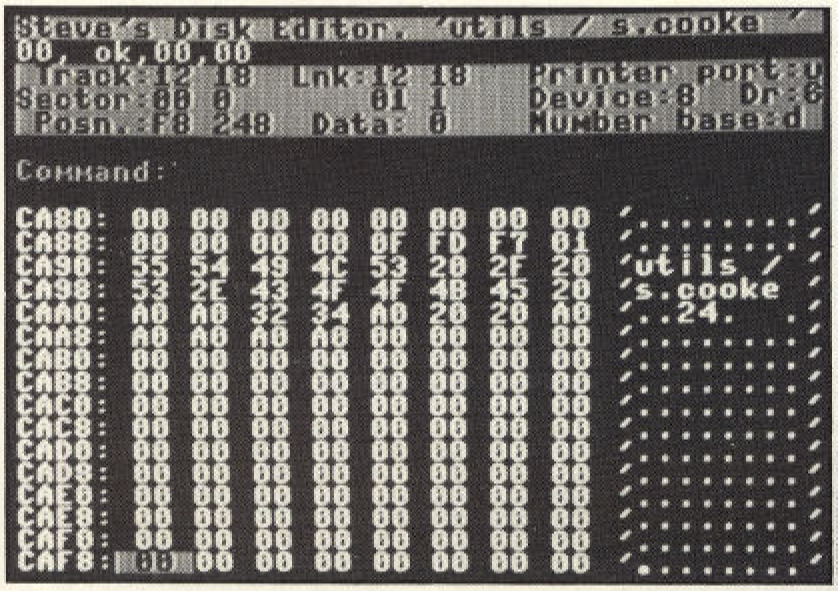
We are therefore proud to announce the start of the 'Your Commodore Software Service'. Most of the programs from each issue of the magazine will now be available on a single cassette for a price of just £4.00. We will not be making disks available since they would have to be a lot more expensive and more difficult to post. This shouldn't cause you any problems though as none of the programs will be protected and it will be a simple matter to save the programs to disk yourselves.

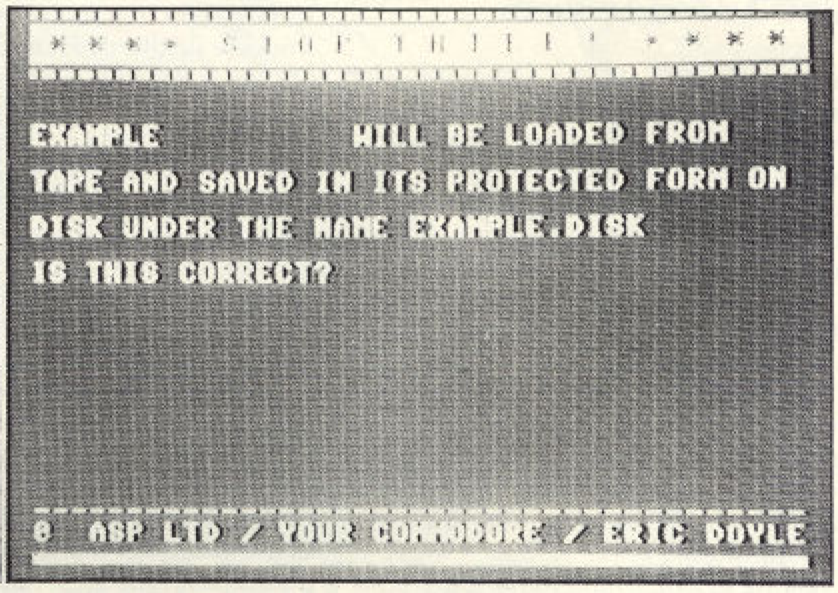
All programs on the cassette will be saved using a tape turbo routine. However, we cannot guarantee that all programs will work correctly with this turbo routine present. We therefore recommend that before you use any of the programs you make a copy of the programs on your own cassette or disk and use this version of the program **not** the original.

This month we are not only going to make available most of the programs from this issue on cassette, we are also









producing a special 'Best of Your Commodore Utilities' cassette. This will contain the following programs from past issues:

MACH 1 TO 4 – Assembler – Nov'85 to Feb'86
STOP THIEF – Feb'86
IN CHARACTER – Character Editor Nov'85
MOB MAKER – Sprite Editor – Mar'85
DISK EDITOR – Jan'86

Unfortunately, instructions for these programs will not be available with the cassettes so you will have to refer to back issues of the magazine for them.

N.B. Only the programs for the C64/C128 will be available on cassette. We are looking at the C-16 and other machines at the moment and these may be available shortly.

The Music Master program will not be available on this month's cassette as it is only available from Wizard Development Ltd.

ORDER FORM

PLEASE COMPLETE IN BLOCK CAPITALS
All cheques/postal orders should be made payable to Argus Specialist Publications Ltd.
I would like to order the following cassettes: (please tick box.)

MARCH '86	CMA86-1 £4.00	
UTILITY SPECIAL	CMA86-2 £4.00	
APRIL '86	YCAPR86 £4.00	

I enclose a cheque/postal order for £..................................

NAME ....

ADDRESS ....

....

....

POSTCODE..................................

All orders should be sent to:
Your Commodore Softw·re, Readers Services, Argus Specialist Publications, Wolsey House, Wolsey Road, Hemel Hempstead, Hertfordshire HP2 4SS.

COMMUNICATION

David Janda keeps you in touch with the services which keep you in touch.

I HAVE HAD QUITE A FEW MAILBOXES from subscribers on Micronet and Compunet about the subject of costs. So, besides the usual banter and news, I would like to express in general terms and views of the subscribers, Micronet and Compunet on this sensitive subject.

Value Added Services

What are they? Well, Micronet and Compunet are Value Added Networks (VANs). That is, besides the 'usual' services such as mailboxing, system information such as directories and so on, they also provide the subscriber with other goodies such as Micronet's Starnet, or Compunet's Multi User Dungeon (MUD). This type of goodie is called a Value Added Service (VAS).

The type of mailbox I have been getting goes along the lines of; 'Why should I pay X amount for such-and-such when I've already paid Y pounds subscribing to the thing in the first place?' A fair question.

Talking to Micronet's mega-star Public Relations Manager, Peter Probert gave me M'Net's side of the story. Peter explained that when M'Net first started it had no VASs. The info on the system was supplied as part of the subscription. However, the more elaborate features such as Starnet, Gallery and the Daisychats cost a lot to develop and maintain as far as administration is concerned. Micronet is not in business to rip-off its subscribers, but neither is it a registered charity.

M'Net's Technical Guru and VAS manager, Mike Brown, told Your Commodore; "The extremes are that we could have no subscription and charge for every page looked at, or we could have a very high subscription and no charge for any of the services. We prefer to strike a balance between the two."

As far as users are concerned, they can use the regular 'free' facilities of M'Net such as the news and user bases, and have the choice of using the services that cost. Development wise, it's easy to think that M'Net would be tempted to spend more time/effort in developing VASs, but this would not appear to be the case. Bizznet, AIQ, the Sunday mag and the Amiga section are new or newish sections which do not cost anything.

Jane Fairbank, Editor of Compunet, told Your Commodore much the same thing. Jane emphasised the calories it takes to develop a new service; "Something like Party-line takes a lot of time and money to develop." she said. Having used the MUD and Party-Line links, I can believe it.

But what about the costs of a VAS? Well, Micronet charges 99p per Gallery frame and 10p to edit a frame. Starnet costs 99p to register and 25p per move. The Round Britain Race costs 1/2p per move and Daisychat 2p per message. Compunet charges £1.75 per hour for MUD and £1 per hour for Party-line. The one-off link charge for MUD or Party-line costs 50p. You can buy a MUD or party-line permanent link for £7.99 and £5.95 respectively.

Needless to say, both Micronet and Compunet justified the costs when questioned; the question is do you think they are justified? Please write or mailbox and let me know.

Here is the News

Micronet — I have raved on about Starnet, the Net's interactive Space Game before in this column, and I heard that the Net is considering another interactive game along the lines of Stocks and Shares. The idea being that you buy and sell according to rules, form cartels and so on. I hope the idea becomes reality, sounds good. At the time of writing, BT's version of MUD is undergoing Beta tests, I have been informed that once the game is up and running proper, there will be a route to MUD from Micronet.

A couple of new areas on the Net should be active by the time you read this. First, the Amiga will have its own area where you will find articles about it. A Sunday Supplement type of magazine will appear in the news index over the weekends. I'm not sure of the contents, but a regular Sci-Fi story and a column written by some voyeur (a what?) of the computer industry is promised.

Finally, it is hoped that the Quickchat channels will be fully debugged by the end of this month. Each Quickchat window will be split into four parts, each showing a message and being updated before your very eyes. An option to back-track is also promised.

Compunet — Jane Fairbank confirmed that CNET is considering a home banking service, but would not give any more away. She also said that a Telex link is possible, but again would not confirm or let on. What is confirmed is a new area for the C128 aimed at the more serious user. A major re-grouping of directories and tidying up should have been completed by now, and faster response times for the whole of the system should be the norm.

Having difficulty remembering all those GOTO numbers? Well, an alpha-numeric GOTO on the Duckshoot is now implemented. Instead of entering 107701, you should be able to enter NEW, or NEWS or MUD or whatever! Users who operate popular areas will most likely find that editorial assistance and financial help (free uploads) are being offered to them — about time too.

Party-line Mk II is now operational with separate 'rooms' enabling you to see who is in them. P-line 2 has a D&D *DICE command allowing users to play D&D. Contact Yeti on AL2 if you want to organise a game.

Eyeball

On M'Net? Then try the new Quickchats at *811. Check out the new Sunday magazine and do youself a favour and register as a Star Captain on Starnet *800030000. C'Netters can check out W.E.E.T. (155806) for a great magazine by SJ2. If you're into comms news, CNET and hard facts reporting, the Dave Martin's MEGA SCENE at 110832 is guaranteed to interest you.

Next month, more lies and a refreshing look at the Bulletin Board scene in and around the UK! Until then, please drop me a line, rope or whatever on Prestel: 919992677 and Compunet ID D.JANDA.

Compunet Answers

Jane Fairbanks of Compunet puts CNet's point of view.

The present Party-Line link is a very sophisticated piece of specially written software, with windowing, scrolling, save and print facilities.

It was offered at 10p as an introductory price, but has now gone up to 50p but at the same time a free dumb teletype link has become available. This is a standard piece of software and does not have any of the facilities described above. But it means people will have the option of using the advance link or the basic one.

We have the same situation in MUD. Until recently, that was only useable with a dumb teletype link, which was and still is free. But a similarly advanced terminal is again available at 50p for those who chose to use it.

In both cases, people intending to use the advanced links frequently can buy and subsequently pre-load, or load on-line, the link as a piece of software; this naturally means they do not have to pay the 50p.

The present P-Line link now costs 50p and the free terminal is now available. If you compare the two it will be abundantly obvious what people are paying for with the advanced link. P-Line itself will soon gain new facilities, notably 'rooms' and a bulletin board.

CORNER

· AIR · COMBAT · EMULATOR ·

· BY CASCADE GAMES ·

1-3 Haywra Crescent Harrogate

£10.95
INC. FREE WATCH AND POSTAGE

THE FASTEST GAME ON TWO WINGS! AND A FREE MULTIFUNCTION CALCULATOR WATCH



The AIR COMBAT EMULATOR, packed with exciting features, is by far the fastest, most realistic flight simulator available. For the first time on a home computer, feel what it is **really** like to fly a high performance military jet and engage tanks, missiles, ships, helicopter gun-ships and two types of cunning intelligent enemy aircraft.

Featuring over 300 sprite images, ACE has the smoothest, most detailed cockpit view of hills, trees and other ground objects ever achieved thanks to specially developed Dynamic Graphic Technique (D.G.T.©) allowing sensational high speed **solid** 3D graphics.

In combat, your head will spin as you struggle with the help of the advanced avionics which include a rear-view camera and radar, to stay alive long enough to do some damage to the enemy.

The unique two seat flying option (pilot, navigator/weapons man) and Synchro Speech© which passes vital aircraft condition information to the flight crew without freezing the action combined with dynamic tactical map, take offs and landings, summer, winter and night-time flying and a remarkably realistic in-flight refuelling option makes ACE the ultimate AIR COMBAT EMULATOR.

ACE SCENARIO

A vast invasion fleet has landed on your southern shores and discharged scores of tanks supported by helicopter gun-ships and protected by mobile surface to air missiles. Squadrons of enemy fighters give the enemy forces air cover. Advancing ruthlessly they are conquering your homeland, facing no opposition.

You are the last fighter pilot with only three A.C.E. Mark 2.1 multi role All Weather All Terrain AWAT© combat aircraft. Based at an allied airbase, your task is to attack and destroy the numerically superior enemy land and air forces and drive them from your shores. The final stage of the conflict, having already destroyed the enemy land forces and shot down their air force, is to sink the enemy fleet as it evacuates its defeated army.

ACE – EXPERIENCE IT NOW

Every computer game fan will love ACE – just look at these features:-

- 100% optimized machine code uses all available memory.
- Featuring Dynamic Graphic Techniques (D.G.T.©)
- The smoothest, fastest and most detailed cockpit view of any flight simulation.
- Views of hills, trees, tanks and other ground objects. Aerial combat, ground attack and naval attack.
- Synchro Speech© ie. the speech does not freeze the action.
- Two seat option – fly and fight with the aid of a friend.
- Sensational 3D solid (not boring wireframed) graphics.
- High score saving system.
- AWAT© operation (All Weather All Terrain).

* Note due to memory limitations ACE on the C16 and VIC20 (+8K Ram) do not have any ground objects.

FREE 20 KEY MULTIFUNCTION CALCULATOR WATCH
• 12/24 Hour Time Mode • Alarm Mode • Calendar Mode • Calculator Mode



Screen shots are for CBM 64 version. Other versions may vary.

ACE – EXPERIENCE IT NOW!

FREE!!!

Order ACE today and receive this fantastic
20 KEY MULTIFUNCTION CALCULATOR WATCH

ABSOLUTELY FREE!!!

PRICE **£10.95 (CBM 16 and VIC 20 £9.95)**

Postage FREE in U.K. Add £1 for overseas orders.
Allow 28 days for delivery.
I enclose cheque/postal order for (£..............................) made payable to Cascade Games Ltd.
Or through any Post Office by Transcash (Giro No. 655 6655).
For even faster ordering when charging to Access, Barclaycard and Trustcard Visa use our 24 hr service (Tel. 0423 504663).

ENTER CREDIT CARD NO. IN BOX:-

VISA

☐ C= 64 ☐ C= 16 ☐ C= VIC 20 (with 8K RAM expansion)
☐ C= PLUS 4 ☐ C= 128

Please tick box
NAME
ADDRESS

POSTCODE
COUNTRY

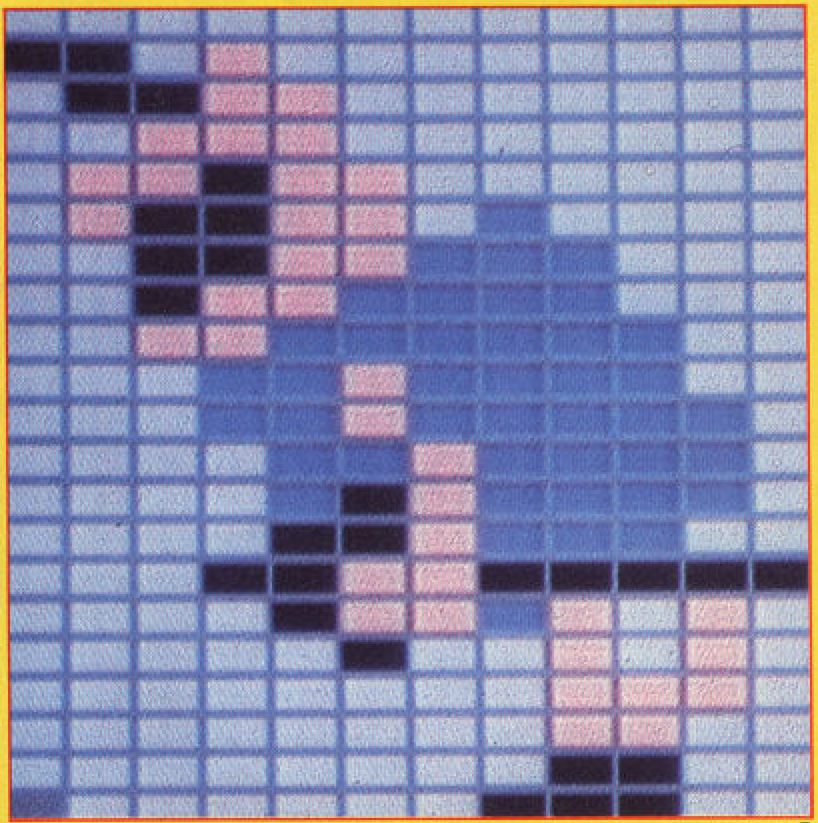


Please complete and send to:-
Cascade Games Ltd.
1-3 Haywra Crescent
Harrogate HG1 5BG. England.
Tel. (0423) 504663
Registered No. 1755554

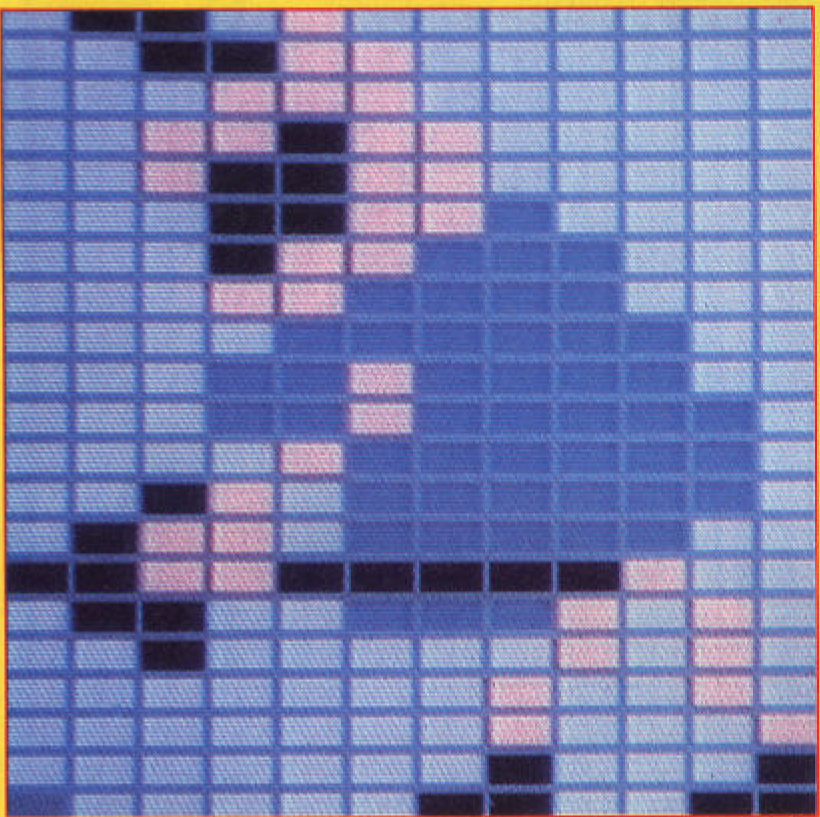
cascade

YC4

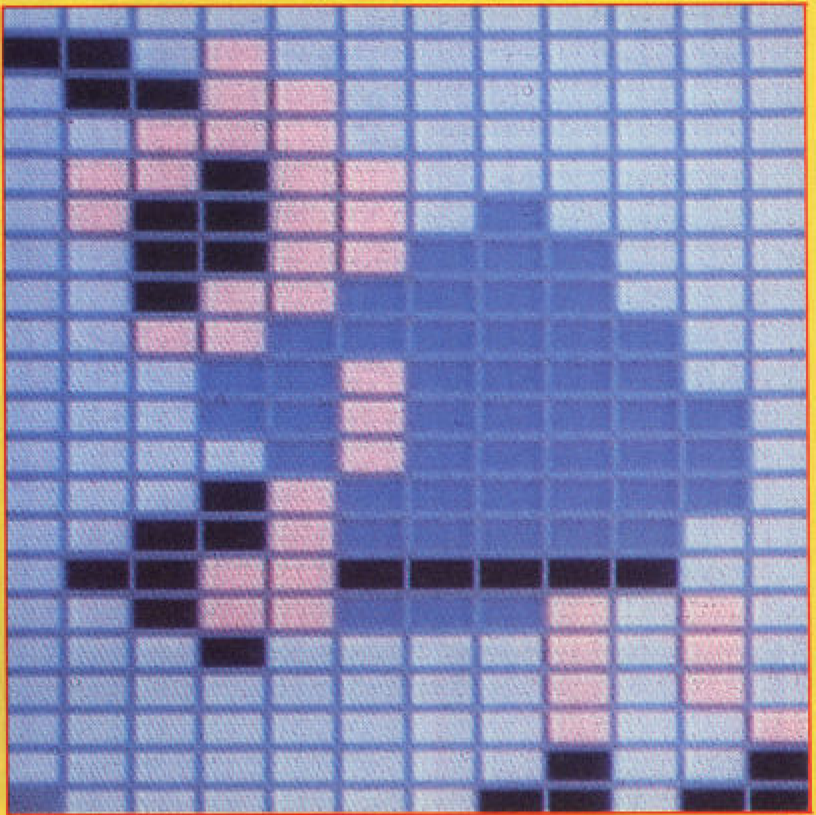
SPRITE IDEAS



1



4



5

When you are designing a game one of the longest jobs is designing the sprites. If you are good at art then fine, if not your next monster will probably end up looking like a square box with legs.

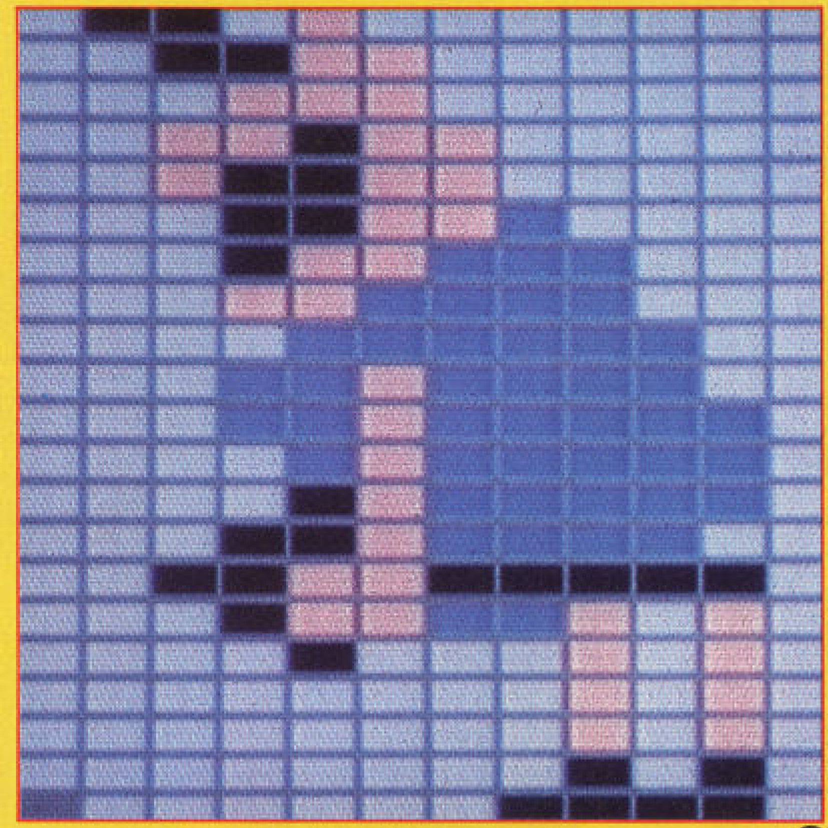
Now, Your Commodore comes to the rescue once again with Sprite Ideas. If you have designed any sprites for games and you don't mind other people seeing your masterworks then why not send them into us. Each month we will be offering £10 for the best entries.

Your sprites can be anything at all (within reason), if you've designed a series of animated characters then send in the lot. We'd love to have a look at them.

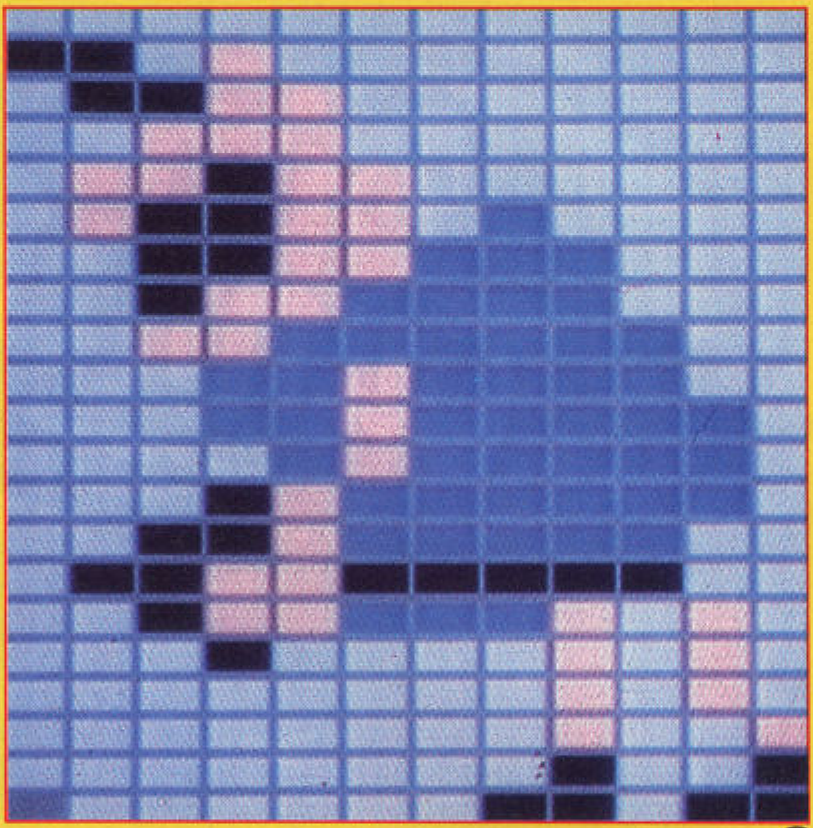
So, next time you are after an Ogre to put in your new game, have a look in this section of the magazine and you may find just what you are looking for.

```
5 FORX=0TO1000:READA:POKE(64*245)+X,A:NEXT
10 VIKING 1
11 DATA000,000,000,163,000,000,043,192
12 DATA000,015,192,000,062,240,000,058
13 DATA241,000,010,245,064,011,213,064
14 DATA015,085,080,001,117,080,001,117
15 DATA084,000,093,084,000,109,084,000
16 DATA173,080,002,190,170,000,189,204
17 DATA000,032,204,000,000,252,000,000
18 DATA240,000,000,160,000,002,160,000
20 VIKING 2
21 DATA040,192,000,010,240,000,003,240
22 DATA000,015,188,000,014,188,000,002
23 DATA189,000,002,245,064,003,213,064
24 DATA000,085,080,001,117,080,001,117
25 DATA084,000,117,084,000,181,084,002
```

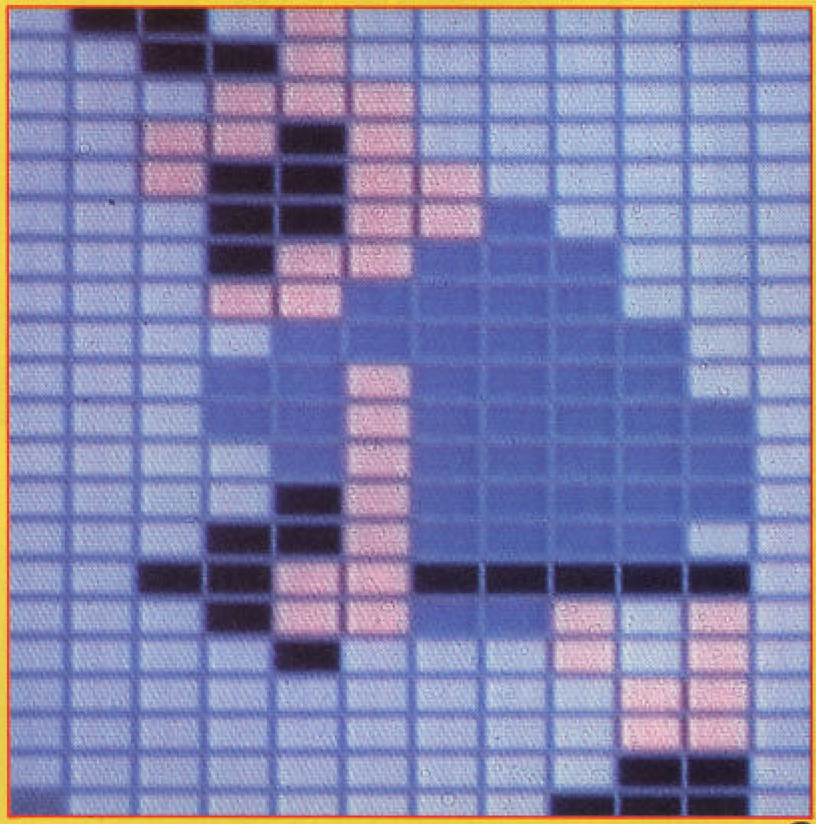
All sprites by — Simon Deacon, Esher, Surrey



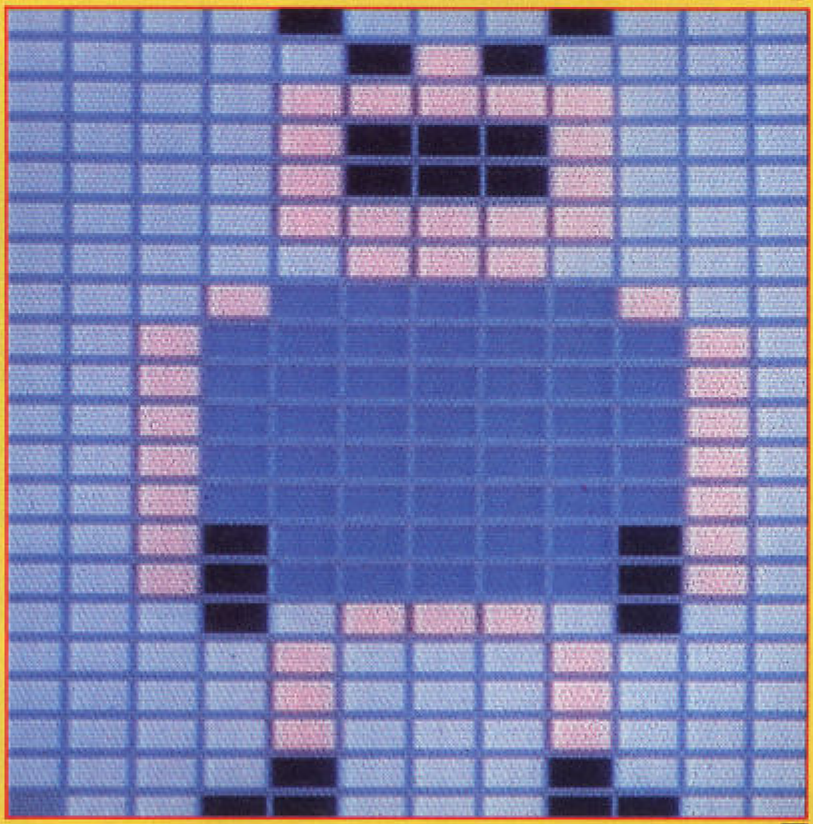
2



3



6



7

```
26 DATA181,080,010,250,168,002,245,204
27 DATA000,128,204,000,000,204,000,000
28 DATA204,000,000,136,000,002,168,000
30 VIKING 3
31 DATA000,000,000,163,000,000,043,192
32 DATA000,015,192,000,062,240,000,058
33 DATA241,000,010,245,064,011,213,064
34 DATA015,085,080,001,117,080,001,117
35 DATA084,000,117,084,002,213,084,010
36 DATA213,080,043,234,160,011,213,204
37 DATA002,000,204,000,000,204,000,000
38 DATA195,000,000,130,000,002,138,000
40 VIKING 4
41 DATA040,192,000,010,240,000,003,240
42 DATA000,015,188,000,014,188,000,002
43 DATA189,000,002,245,064,003,213,064
44 DATA000,085,080,001,117,080,001,117
45 DATA084,000,213,084,011,021,084,047
46 DATA021,080,175,170,176,040,021,204
47 DATA008,000,204,000,003,012,000,003
48 DATA003,000,002,002,000,010,010,000
50 VIKING 5
51 DATA000,000,000,163,000,000,043,192
52 DATA000,015,192,000,062,240,000,058
53 DATA241,000,010,245,064,011,213,064
54 DATA015,085,080,001,117,080,001,117
55 DATA084,000,117,084,002,213,084,010
56 DATA213,080,043,234,160,011,213,204
57 DATA002,000,204,000,000,204,000,000
58 DATA195,000,000,130,000,002,138,000
60 VIKING 6
61 DATA040,192,000,010,192,000,003,240
62 DATA000,015,176,000,014,188,000,002
63 DATA189,000,002,245,064,003,213,064
64 DATA000,085,080,001,117,080,001,117
65 DATA084,000,117,084,000,181,084,002
66 DATA181,080,010,250,168,002,245,204
67 DATA000,128,204,000,000,060,000,000
68 DATA060,000,000,040,000,000,168,000
70 VIKING 7
71 DATA000,128,128,000,046,000,000,255
72 DATA192,000,234,192,000,234,192,000
73 DATA255,192,000,063,000,003,085,112
74 DATA013,085,092,013,085,092,013,085
75 DATA092,013,085,092,013,085,092,014
76 DATA085,108,014,085,108,002,063,032
77 DATA000,192,192,000,192,192,000,192
78 DATA192,000,128,128,002,128,160,000
```



Explore the depths of history, travel to the far distant future, assemble the mystical map from the seven ages of mankind and become King of the Gnomes
'Probably the best arcade adventure ever produced in the U.S.A.'
Time Tunnel
Intergalactic Spaceship 3556 – You are now in the far distant future, guide your craft to the alien planet but don't venture onto the surface for too long!!!
The Black Hole 9999 – After travelling through time, assemble the map and your incredible task is almost complete...
California Gold Rush 1849 – Deadly scorpions, beastly bats, and numerous nasties are on the attack as you search the cavernous gold mines of California.
The Gnome Mansion – the mysterious location of your fantastic time machine. Assemble the final parts of your craft and begin your epic voyage to past and future.
Mythological Greece 43BC – A confrontation with Medusa, a fight to the death with only your magical shield for protection.
Stone Age 9600BC – Explore the dark and dangerous caves of prehistoric times, you'll need strength and cunning to reach your goal.
Colonial Salem, MA 1692 – Hubble, bubble, toil and trouble, mix a ghastly witches brew and you'll win through, but watch out for the flying broomstick.
£14.95 DISK
£9.95 TAPE
U.S. GOLD All American Software
COMMODORE 64/128
U.S. Gold Limited,
Unit 10, Parkway Industrial Centre, Heneage Street, Birmingham B7 4LY. Telephone: 021-359 8881

Teacher's Pet

Margaret Webb has been inside her C64 to bring you this month's column.

IN RECENT MONTHS I'VE DEVOTED each article to specific areas such as music, adventures and graphics. This month, just for a change, I plan to describe a handful of miscellaneous items of software which cover different topics.

The first item, *D-Bug* (Ariolasoft), appeared to be simply a game but it transpires that it is a pretty good piece of educational material which teaches a little about the structure of computers. The idea is quite novel. You are attempting to play a game called *Gotcha* in a software generated computer. This is a tactical game in which you endeavour to force the opponent to take your pieces on a grid playing area. The only problem is that the 'computer' keeps developing bugs. Using the symptoms shown in the video output or the sound, you must dive into the interior of the computer and rectify the fault. This entails replacing or nudging an awkward chip or other appropriate action. The game is both colourful and noisy and combined with the comprehensive instructions gives a decent introduction to computer architecture. Probably best suited to the nine to 13 age group.

A frequent feature of the software market is the appearance of 'licensed' software based on films. Some appears to be quite good, some of it pretty rough. Bearing this is mind, it was with some trepidation that I loaded *The Neverending Story* from Ocean. Fortunately, my fears were generally unfounded. This is a text/graphics adventure supplied as a two cassette package. The game is split into three sections and you may only progress once you have completed a section. OK, so what has this to do with education, you all shout. Well, if you had read my article on adventures a couple of months ago, you would know the answer.

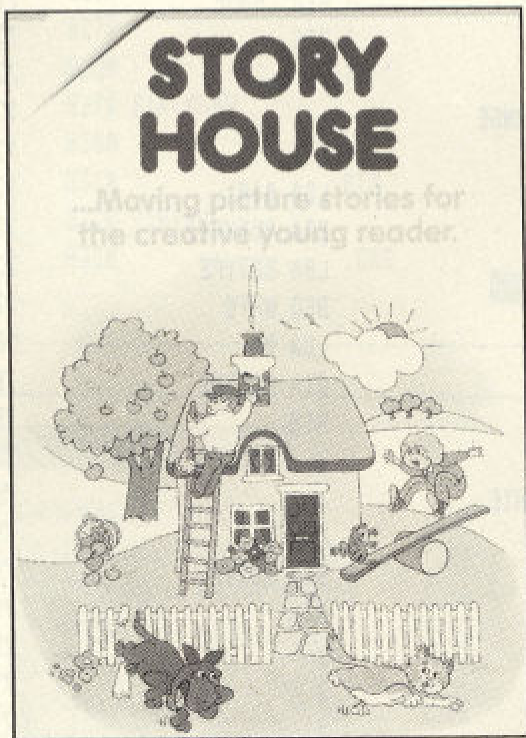
So what's it like? Well, my better half, who's a bit of an adventure freak (well it keeps him off the streets!), thinks it's the prettiest game for yards around. I'm inclined to agree with him. The illuminated 'gothic' text style is nicely enhanced with some pretty multicolour high resolution pictures. These pictures not only show your location but also what you're carrying. When you start to play, however, the content doesn't quite live up to the graphics. The apparent lack of interaction with the other characters (they kept running away!) and the limited vocabulary tend to make the play a little irritating. The story line and the graphics are certain, however, to appeal to younger players.

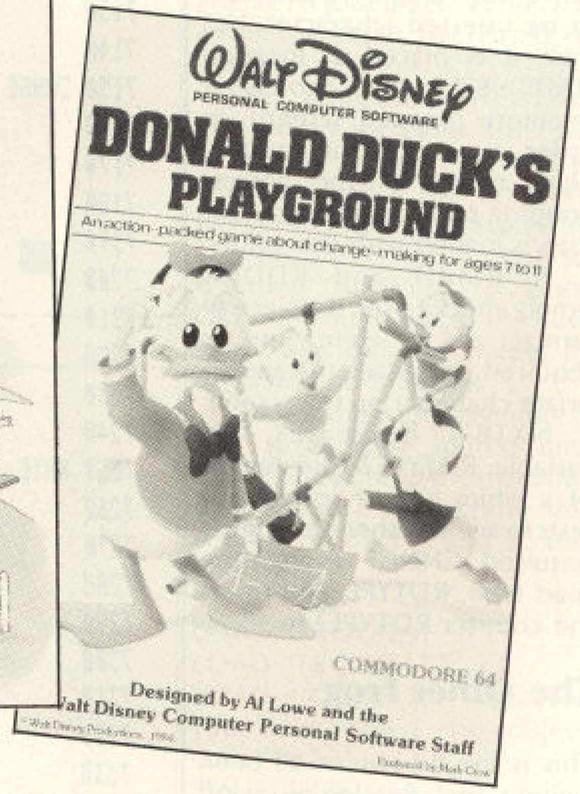
Most of you will have seen the US Gold trademark on a wide range of American software. Our friends at US Gold are now marketing a range of Walt Disney material on a label called Kids. As far as I'm aware, three titles are now generally available. The only one I've seen to date is *Donald Duck's Playground* (the others involve Winnie the Pooh and the Muppets – I hope to see these soon. The game involves the erection of an adventure playground for Donald's nephews. As with most of life, nothing comes free, and you must earn cash in order to purchase the component parts of the playground. Four jobs are available to Donald each utilising a different skill. The airport job involves matching letters and at the toy shop you must match shapes. Unloading the fruit lorry is more of a test of dexterity and controlling the train system requires logic and lateral thought. Some may object to the use of dollars and cents but since we also use a decimal system, this isn't too much of a problem.

Not surprisingly, the graphics are to a high standard and the use of sound is adequate. Overall, a nice product which is ideal for the pre-school/infants age group.

The final piece of software, *Story House* – Creative Sparks, is also aimed at the younger age group. The idea is to help children to read by allowing them to build up illustrated stories. The program is multiple choice and allows the selection of the characters in the story, their names and other details. The game can be replayed at regular intervals and can be saved on cassette for repeat performances. The graphics are quite pretty with the characters appearing as animated sprites. (the antics of the cat, frog etc. illicited great jollity from my five year old son). Again, ideal for pre-school children (although the package quotes five to nine years) and will keep kids quiet whilst mum gets on with whatever she does.

Well folks, thats all for this month. I do hope that I've given you some ideas to help you keep your children busy and learning.





There's more about frogs and Frenchmen in this month's installment of our exciting arcade game series. By Daryl Bowers.

FIRSTLY, IN THIS MONTH'S exciting episode, we've got some more sprite data to type in. This is entered using the routine which I gave you in part one – the start address being 13312 and number of bytes being 1024. This should be saved as FROGDAT2. This data contains all the rest of the sprites required for the game.

Now on to the exciting bit! Line 1530 contains a jump to the bicycle printing and moving routine, CYCLE, which we will come to later. The first routine, however, is an extension of the routine ROADFILL.

The variable CRATER contains the number of characters of a puddle left to print, or, if zero, it indicates that there is no puddle and that a normal piece of road should be inserted on to the right hand side of the screen. If a puddle is to be inserted (character 102) then it is placed at location $06D0 + $2F in the video matrix (memory mapped screen). In order to make the puddle blue (colour 6) the corresponding position in the colour map is loaded with this value.

NORMBIT and RDDRK simply check to see if a piece of tarmac or a white line is required, and place the appropriate character on the screen.

NXTBIT decreases the variable RDTYPE2 (the length of a white line or space) and tests to see whether a change is required. CHNGE reverses the road type, RDTYPE, and resets the counter RDTYPE2 to 10.

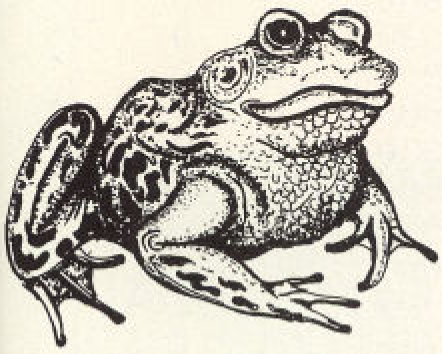
The Other Frog

This is the bit you've all been waiting for! Bearing in mind the fact that I'm not a graphic designer and no good at art, I hope you appreciate the design of the Frenchman. Well, I thought it was quite good! But I digress. Back to the code.

PROGRAM: FROGGY PT4

```
1530          JSR CYCLE
6820          ;
6830          LDA CRATER
6840          BEQ NORMBIT
6850          DEC CRATER
6860          BEQ NORMBIT
6870          LDA #102
6880          STA $06D0+$27
6890          LDA #6
6900          STA $DAD0+$27
6910          JMP NXTBIT
6920          ;
6930          ;
6940 NORMBIT
6950          ;
6960          LDA RDTYPE
6970          BEQ RDDRK
6980          LDA #160
6990          STA $06D0+$27
7000          LDA #1
7010          STA $DAD0+$27
7020          JMP NXTBIT
7030          ;
7040 RDDRK
7050          ;
7060          LDA #32
7070          STA $06D0+$27
7080          ;
7090 NXTBIT
7100          ;
7110          DEC RDTYPE2
7120          BEQ CHNGE
7130          RTS
7140          ;
7150 CHNGE
7160          ;
7170          LDA #10
7180          STA RDTYPE2
7190          LDA RDTYPE
7200          BEQ WHTE
7210          LDA #0
7220          STA RDTYPE
7230          RTS
7240          ;
7250 WHTE
7260          LDA #1
7270          STA RDTYPE
7280          RTS
7290          ;
7300          ;
7310          ;
7320          ;
7330          ;
7340 CYCLE
7350          ;
7360          ;
7370          ;PRINT SPRITES
7380          ;
7390          LDA #154
7400          STA $D007
7410          STA $D009
7420          LDA #121
7430          STA $D00B
7440          LDA #164
7450          STA $D005
7460          LDY #3
7470          LDX #$0A
7480 LOOP19
7490          LDA FRENCH,Y
7500          STA $D000,X
7510          DEX
7520          DEX
7530          DEY
7540          BPL LOOP19
7550          LDA FRSPRT
7560          STA $07FA
7570          LDA #208
7580          STA $07FB
7590          LDA #209
7600          STA $07FC
7610          LDA #210
7620          STA $07FD
7630          ;
7640          ;
7650          ;
7660          ;
7670          DEC FRDEL
7680          BNE NOBIKE
7690          LDA #200
7700          STA FRDEL
7710          INC FRSPRT
7720          LDA FRSPRT
7730          CMP #217
7740          BNE NOBIKE
7750          LDA #211
7760          STA FRSPRT
7770 NOBIKE
7780          LDY BIKEPOS
7790          DEC BIKEDEL
7800          BNE BIKENOMV
7810          LDA BIKEMV+1,Y
7820          STA BIKEDEL
7830          DEC BIKEDIS
7840          BEQ NEWDIR
7850          LDA BIKEMV,Y
7860          BEQ BIKENOMV
7870          BMI BACKBK
7880          LDX #3
7890          LDA FRENCH
7900          CMP #110
7910          BEQ BIKENOMV
7920 LOOP20
7930          INC FRENCH,X
7940          DEX
7950          BPL LOOP20
7960          JMP BIKENOMV
7970          ;
7980 BACKBK
```

FROGGY





```
7990            LDX #3
8000            LDA FRENCH
8010            CMP #20
8020            BEQ BIKENOMV
8030 LOOP21
8040            DEC FRENCH,X
8050            DEX
8060            BPL LOOP21
8070            JMP BIKENOMV
8080            ;
8090 NEWDIR
8100            LDA BIKEPOS
8110            CLC
8120            ADC #3
8130            STA BIKEPOS
8140            TAY
8150            LDA BIKEMV,Y
8160            CMP #$FE
8170            BNE GETDIR
8180            LDA #0
8190            STA BIKEPOS
8200            ;
8210 GETDIR
8220            LDA BIKEMV+1,Y
8230            STA BIKEDEL
8240            LDA BIKEMV+2,Y
8250            STA BIKEDIS
8260            ;
8270            ;
8280            ;
8290 BIKENOMV
8300            ;
8310            RTS
8320 FINISH
8330            .END
```

First, we set up the Y co-ordinates of the four sprites. As the Frenchman does not move up or down the screen, these will remain constant, and could even be set up at the start of the program and never altered.

Next we have a short loop that reads the X co-ordinates from the table FRENCH and inserts these into their appropriate positions in the VIC Input/Output area.

Last, as far as printing the sprites goes, we place the sprite definitions into the sprite Data Pointers. Only one of these changes (the legs) and the current value of this is stored in FRSPRT.

PROGRAM: SPRITE DATA

```
13312 :0000000000000000   CH : 0
13320 :000000000003E000   CH : 227
13328 :01800003FF000400   CH : 391
13336 :070800018D0002020  CH : 311
13344 :0040500140900283   CH : 486
13352 :6B7487088882C934   CH : 885
13360 :4011244016882020   CH : 403
13368 :7018C000070000FE   CH : 589
13376 :0000000000008000   CH : 128
13384 :0060000010000010   CH : 128
13392 :0000F00000100000   CH : 256
13400 :37000058C000B020   CH : 543
13408 :0050100050100088   CH : 328
```

Lines 7670 to 7760 concern the animation of the legs. The variable FRDEL is a counter which decreases from 200 to zero to create a delay between movements. When it reaches zero FRSPRT (the leg sprite number) is increased. If the value has reached 217, the last leg sprite, it is reduced to 211, the first leg sprite. This creates a loop displaying all sprites in sequence.

The remaining code is concerned with the movement of the bike. The table BIKEMV controls the bike movement. Each entry is three bytes long and contains the following information:

Byte	Meaning
1	Direction of movement (255/0/1)
2	Speed of movement
3	Duration of movement (distance moved)

BIKEPOS contains the offset to the current entry in the table. BIKEDEL contains the delay until the next movement, and is decreased to see if a new movement is required. If it is, BIKEDEL is restored to the value found in byte two of the table entry and the Duration of movement is decreased. If this has reached zero then a jump is made to NEWDIR. Next the current direction is found and FRENCH (discussed earlier) is either increased, decreased, or left the same.

This brings us to NEWDIR. This simply increases BIKEPOS to point to the next entry, checks to see if the last entry has been reached (indicated by $FE in the table) and if so resetting BIKEPOS to zero. GETDIR simply places the new speed and duration values in BIKEDEL and BIKEDIS.

To test the code, assemble it and save both source and object code. Reset the machine (making sure you know the address of START), and do the following:

LOAD "FROGDAT",8,1
LOAD "FROGDAT2",8,1
LOAD "CODE (or whatever you have called the object code)",8,1
SYS (START)

A joystick is required in PORT 3.

Next month we will introduce the Bird and the Fly! Can you wait?

```
13416 :0800860800800800 CH : 286
13424 :4010004010002020 CH : 224
13432 :0018C000070000BF CH : 414
13440 :0000000000000000 CH : 0
13448 :00028000AA000003 CH : 303
13456 :4000035000037000 CH : 262
13464 :03400000400003F0 CH : 374
13472 :0001500003540001 CH : 169
13480 :540003D500015500 CH : 386
13488 :03F14001506003F0 CH : 728
13496 :2001500003F00001 CH : 357
13504 :0AA00002A80008A8 CH : 516
13512 :000A28000A28000A CH : 110
13520 :28000A28000A3F00 CH : 163
13528 :0A3F000A00000A00 CH : 93
13536 :000A00000A00000F CH : 35
13544 :C0000FC000000000 CH : 399
13552 :0000000000000000 CH : 0
13560 :0000000000000040 CH : 64
13568 :0A80000AA00002A8 CH : 478
13576 :0008A8000A28002A CH : 268
13584 :2800282800A82800 CH : 328
13592 :A02800FC3F00FC3F CH : 830
13600 :0000000000000000 CH : 0
13608 :0000000000000000 CH : 0
13616 :0000000000000000 CH : 0
13624 :00000000000000FE CH : 254
13632 :0A8A000AA2000AA8 CH : 498
13640 :0002A80000A8000A CH : 348
13648 :28002A2800382800 CH : 218
13656 :3C28000F28000328 CH : 198
13664 :00002800003F0000 CH : 103
13672 :3F00000000000000 CH : 63
13680 :0000000000000000 CH : 0
13688 :00000000000000BF CH : 191
13696 :028800028A00028A CH : 418
13704 :00028A00028A0002 CH : 282
13712 :8A00028A00028FC0 CH : 615
13720 :028FC00280000280 CH : 597
13728 :0002800002800003 CH : 263
13736 :F00003F000000000 CH : 483
13744 :0000000000000000 CH : 0
13752 :0000000000000001 CH : 1
13760 :0A80000A88000288 CH : 422
13768 :000A88000A28002A CH : 238
13776 :2800282800A82800 CH : 328
13784 :A02800FC3F00FC3F CH : 830
13792 :0000000000000000 CH : 0
13800 :0000000000000000 CH : 0
13808 :0000000000000000 CH : 0
13816 :0000000000000040 CH : 64
13824 :0AAA000AAA00002A CH : 402
13832 :0000A80002A0000A CH : 340
13840 :88002A2800382800 CH : 314
13848 :3C28000F28000328 CH : 198
13856 :00002800003F0000 CH : 103
13864 :3F00000000000000 CH : 63
13872 :0000000000000000 CH : 0
13880 :00000000000000FE CH : 254
13888 :0000000000000000 CH : 0
13896 :0000000000000000 CH : 0
13904 :0000000000000000 CH : 0
13912 :0000000000000000 CH : 0
13920 :00000000000000AA CH : 170
13928 :0AA0FEAAAA2AA96A CH : 1081
13936 :AA5AA6EAAAAA3EAA CH : 1232
13944 :6AAAEAAFFF3FF0BF CH : 1434
13952 :0000000000000000 CH : 0
13960 :0000000000000000 CH : 0
13968 :0000000000000000 CH : 0
13976 :0000000000000000 CH : 0
13984 :0000000000000000 CH : 0
13992 :00008A0000A98000 CH : 435
14000 :6AA000AAB000B9C0 CH : 989
14008 :00CF000000000001 CH : 208
14016 :0000000000000000 CH : 0
14024 :0000000000380000 CH : 56
14032 :4400004400004400 CH : 204
14040 :033E0007FF8007FF CH : 717
14048 :80033E0000000000 CH : 193
14056 :0000000000000000 CH : 0
14064 :0000000000000000 CH : 0
14072 :0000000000000040 CH : 64
14080 :0000000000000000 CH : 0
14088 :0000000000000000 CH : 0
14096 :0000000000000000 CH : 0
14104 :000000033E0007FF CH : 327
14112 :8007FF80033E0000 CH : 583
14120 :4400004400004400 CH : 204
14128 :0038000000000000 CH : 56
14136 :00000000000000FE CH : 254
14144 :0000800002000002 CH : 132
14152 :00000A00000A0000 CH : 20
14160 :2A0000280000A800 CH : 250
14168 :00A8003CFFF03FFF CH : 1041
14176 :F07FFF003FFC000C CH : 949
14184 :F000000000000000 CH : 240
14192 :0000000000000000 CH : 0
14200 :00000000000000BF CH : 191
14208 :0000000000000000 CH : 0
14216 :0000000000000000 CH : 0
14224 :0000000000000000 CH : 0
14232 :0000003CABF03FAB CH : 705
14240 :F07FAB003FAA000C CH : 783
14248 :EA00002A00000A00 CH : 286
14256 :000A800002800000 CH : 268
14264 :8000008000002001 CH : 289
14272 :0000000000000000 CH : 0
14280 :0000000000000000 CH : 0
14288 :0000000000000000 CH : 0
14296 :0000000000000000 CH : 0
14304 :0000000000000000 CH : 0
14312 :0000000000000000 CH : 0
14320 :0000000000000000 CH : 0
14328 :0000000000000040 CH : 64
```

The Adventures of Bond — Basildon Bond
Probe Software

Originality 5 · Playability 4 · Graphics 5 · Value For Money 4



BASED ON THE CHARACTERS in Russ Abbott's television show, you play the part of Basildon Bond as you try to rescue Russ from a rival comedy act. To help you in your quest you can also call on the sevices of Cooperman and Blunderwoman.

The game is set in a television studio. In order to rescue Russ, you must go round the rooms searching assorted bits of furniture for jokes and answers which you must then log into your computer. Things, however, are not that simple and there are various hazards to impede you. To start with, you have to complete your task within a certain time limit. Wandering cameras will transport you back to the start and add 10 minutes to your clock. Moving faces will steal your jokes and answers if they touch you.

To help you, you can summon Cooperman, who flies across the screen like a low-level Harrier strike and takes out any cameras that he hits. The graphics look colourful but movement of your character is crude and jerky and I found that playability was poor both with control of the character and lack of addictive qualities.

This game has pinched a lot of its ideas from the excellent Impossible Mission. If you compare Impossible Mission to a James Bond film, then Basildon Bond is about as exciting as Crossroads. **G.R.H.**

Master of Magic
Mastertronic £2.99

Originality 6 · Playability 8 · Graphics 5 · Value For Money 9



WHILST EXPLORING SOME caverns, you get dragged into a pool and transported into a strange world by Thelric, Master of Magic. He wants the Amulet of Immortality and is prepared to send you back to your own world if you get it for him.

Master of Magic is a menu-driven role playing adventure game. Commands can be selected either with a joystick or via the keyboard. Some options lead into sub-menus and as soon as you call up the option list, the game clock stops so that you can ponder your actions at leisure.

Selecting "run" allows you to move around the dungeons. A map of your current position is displayed in the top left-hand corner. Top right is the information box, the centre window displays the menus and the bottom of the screen shows close-up pictures of monsters, objects and treasures that you meet.

Combat can either be with a weapon that you have found or with one of four types of spell at your disposal. Your physical and mental energies decrease during combat although they can be replenished if you find the appropriate potion.

Master of Magic is very easy to play compared to other games of its type and, although not over sophisticated, is very addictive. At only £2.99, it represents excellent value for money.

G.R.H.

Space Doubt
CRL

Originality 6 · Playability 6 · Graphics 7 · Value For Money 6



NO DOUBT ABOUT IT, THIS game has the longest instruction leaflet that I've seen for some time. Not because it's so complex but because the instructions take the form of a cartoon.

Two of the three heroes of the piece are called Lt. Cdr. Sock and Maj. Underpant which probably explains why they are covering the nether regions of space, transporting food to the planet Niblondis.

The journey is long and treacherous so the crew are in suspended animation in the cryofreezer room – until danger strikes! Asteroids pierce the paper-thin shell of the transporter ship and one by one the crew members are revived to face the mundane task of ship repairs. Fortunately for the game player this boring duty is livened up by an invasion of Bogloid aliens.

Bogloids just love converting humans into steaming pools of grunge and it is each crew member's duty to turn the tables on the amoeboid invaders by skillful use of power block busters. These useful weapons can also seal the outer shell of the ship and keep the Bogloids at bay on the outside of the ship while you flush them away on the inside.

Personally I got more enjoyment from the cartoon strip than I did from the game. If you like pure action buy this but if you want a cerebral challenge look elsewhere.

E.D.

Quake Minus 1

Monolith C64 + joystick

Originality 10 Playability 7 Graphics 8 Value For Money 9



BENEATH THE ATLANTIC Ocean lies the Titan power station. This power station is controlled by five 'Titans' or computers: Zeus, Poseidon, Vulcan, Aries and Hermes. Each Titan has its own function and a fleet of mobiles that can travel around the many roads in the complex making sure that everything is running smoothly.

Of course, things don't run quite as smoothly as the designers of the power plant hoped they would and the Robot Liberation Front have taken control of four of the five Titans and their mobiles. Their aim is to sabotage the whole plant so that they can leave a large portion of the world without power.

Only one Titan still remains under government control – Hermes. You are given control of this computer and all of its mobiles. You must capture or destroy all the other Titans before the RLF takes everything over.

Quake Minus 1 is a rather strange mix of an arcade game and a strategy game. Obviously the aim behind the game is to regain control of the complex, but how you achieve it is completely up to you. For example, if you destroy any of the other Titans then their mobiles become useless and may even block the travel of your own mobiles around the system. On the other hand if you manage to capture any of the Titans then all of their mobiles become yours to control.

You can only control one of the mobiles at any one time, this means that you have to think hard about where you want to go. All of the mobiles have different strengths, each mobile can have up to seven different weapons although in practice each one usually has about four. The best thing to do is find out exactly what weapons each mobile has before you use it. It's not worth travelling all of the way to an enemy complex only to find that the mobile doesn't have the necessary weapon to destroy or capture it.

As previously mentioned, the action takes place on a series of roads. At the meeting of roads is placed a junction, controlled either by the enemy computers or you. You may only pass through a junction when it is under your control.

As you travel along the roads you will come across numerous other installations. Some of these will hinder and some help you. For example if you stop beside a factory then you can have repairs made to the mobile. If you stop next to a fuel tank then you will be able to refuel your mobile. If, however, you should happen to run past any rigs or bunkers that are under enemy control then you are likely to get blown to pieces.

It is simple to see what you control and what the enemy controls as your roads appear as yellow through your mobile window or as red if it is controlled by the enemy.

Controlling your mobiles is easy but it does take time to get used to all of the facilities available to you. Transition mode allows you to move up and down the roads. Thrust mode is accessed while you are moving along the road and displays all of the weapons available. When you come to a stop you can rotate your mobile on the road so that you can take pot shots at the installations along the road and there are many other options too.

Each junction on the road has a maximum of six roads leading from it. When you enter a junction you automatically enter junction mode which will allow you to rotate to all possible roads and view the lie of the land.

Probably the best thing to do with this game is give it a go. It is difficult to play well but extremely easy to get started and you can have a great time zapping anything that moves, and anything that doesn't!

The graphics in this game are quite superbly produced.

As you travel along the roads the objects grow in size and whiz past you.

The game does have one very annoying feature however, an extremely monotonous tune, if it can be called a tune. Being fair it does set the atmosphere for the game but it does tend to grate after a short while.

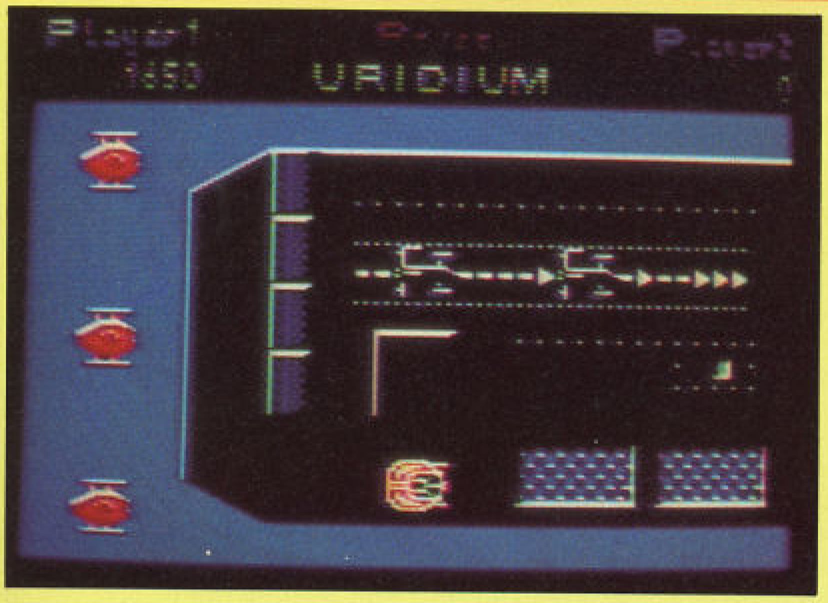
Quake Minus 1 is not a game that will suit everybody. Anyone should be able to get started quite quickly, but you should be prepared to spend a long time getting blown to pieces before you master it.

S.C.

Uridium

Hewson £8.95 cassette £12.95 disk

Originality 7 Playability 9 Graphics 9 Value For Money 8



THE SOLAR SYSTEM IS UNDER attack by Super-Dreadnoughts, gigantic space battleships. The mission is to intercept and destroy each of the Dreadnoughts which are orbiting around the 15 planets in the system.

This exhilarating game from Hewson adds to an already impressive catalogue which includes Gribbly's Day Out and Paradroid. The pseudo 3D metallic graphics are similar to those used in Paradroid but the action is far more frenzied.

As your Manta fighter glides over the battleship's surface, the enemy craft fly in and you must blast them before they get you. Their forms are legion and the higher the class of Dreadnought, the more furiously they will fire. Adding to this problem are the many protruberances from the battleship's surface and the homing missiles which mean instant death if a collision occurs.

A successful attack allows you to land on the surface and then you can collect fuel rods by pressing the fire button at the correct time as a total and a quit sign flash alternately, rather like the gamble feature on a fruit machine. While doing this you must keep an eye on the countdown and select quit before it reaches zero.

This super game is for all those who enjoy quality sound and action mixed with frenzied activity.

E.D.

The Eidolon

Activision £9.95 cassette £14.95 disk

Originality 5 Playability 8 Graphics 9 Value For Money 8



DOCTOR JOSEPH AGON, one of the world's greatest scientists, disappeared over 130 years ago. His relatives quietly closed the mansion where he performed his experiments and it has remained closed until now. You notice that the gates are open and wander into the grounds to have a look round. The front door of the house is also open and as you enter, you notice a strange glow coming from below. You find yourself inexplicably drawn down the steps until you are in Dr Agon's laboratory where you see a strange machine and an old diary.

The Eidolon is the latest game from Lucas Film and features fractal graphics – something which you are likely to hear a lot more of in the coming year. The Eidolon is the name of Dr Agon's strange craft – a craft which is powered by the forces of the mind. As you enter it, you are transported into a series of 3D tunnels, populated by assorted monsters and strange floating spheres.

The spheres are types of floating energy and come in four different types – red, gold, green and blue. They have different functions depending on whether you are catching or firing them. Gold and blue spheres can be found lying around the tunnels and can be collected. Gold ones increase your energy supply and blue ones temporarily suspend the flow of time. You can fire spheres of any colour providing you have sufficient energy simply by selecting the appropriate key one to four on the keyboard). Red spheres act as destructive fireballs, blue ones freeze creatures for a short time and green fireballs polymorph a creature into another which may be more or less dangerous. If you throw a red fireball and it misses its target, it will rebound towards you off the wall. You cannot capture them as such, but if you hit it with another red fireball it will transform into a safe gold sphere which can be re-captured.

The monsters are many and varied. Normally inert, they activate when you approach and must be destroyed. This is usually accomplished by blasting them with a few red fireballs but on lower levels, you may need to fire a blue one first in order to freeze it, or transform it with a green one. If the monster gets too close to your craft, all your energy is depleted and you are returned to the start. When you destroy one of the creatures, you can pick up the glowing energy jewel that it was guarding.

When you reach the end of the particular level on which you are travelling, you come across a large dragon. Normally inaccessible. You can reach it if you are carrying the appropriately coloured crystal. Then, if you destroy the dragon, you are allowed access to the next level. There are seven levels to be explored, the last one being guarded by a many headed dragon. No-one knows what lies beyond. Even Dr Agon doesn't say in his diary.

Control of the Eidolon is very simple – you just push the joystick in the direction you want to travel. The control panel on the screen is very easy to follow. An indicator marked hot and cold tells you how near you are to the dragon. Below that, the gem indicators show which gems have been collected. In the centre of the screen is a clock showing fractions of a minute left. This is surrounded by the fireball indicators – which ones are available and which one is currently selected. Also surrounding the clock is a set of direction indicators. The right hand gauge shows how much energy you have left and below that are two dials showing your current level and the number of whole minutes remaining.

The graphics in the game are stunning. The 3D effects of the tunnels and the animation of the monsters really is excellent. The mathematics of fractal graphics are quite beyond the comprehension of this reviewer but if these are the sort of effects they can produce, then I'm all in favour! There is also a jolly little tune to accompany your craft bobbing up and down the tunnels. Like many games now, the cassette version loads in several parts so there is a lot of tape rewinding to be done – another reason for getting a disk drive.

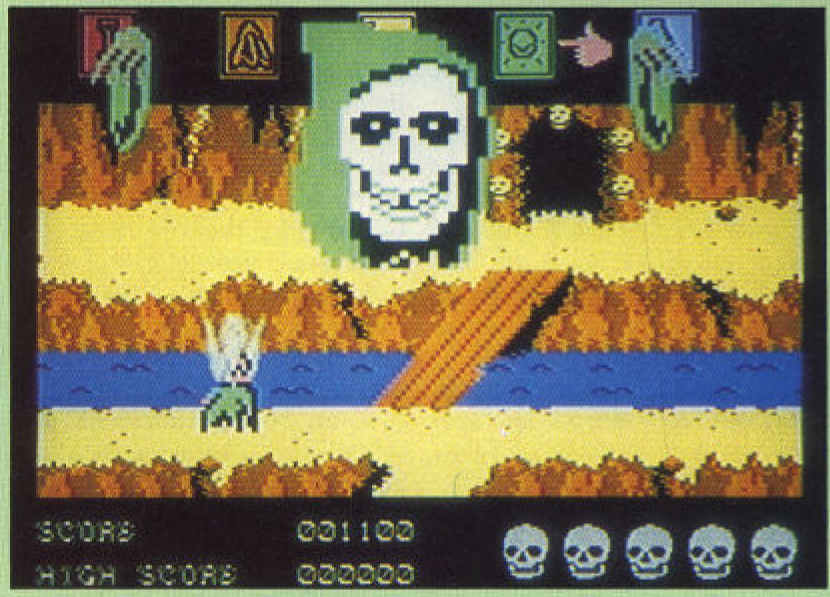
Although basically a 3D maze variant, Eidolon is an extremely well presented, visually attractive, highly playable game.

G.R.H.

Dragon Skulle

Ultimate £9.95

Originality 6 Playability 4 Graphics 8 Value For Money 6



ULTIMATE'S LATEST OFFERING is another in the Sir Arthur Pendragon vein. His quest is to find and destroy the dreaded Skull of Souls and this is why he lands up on a God-forsaken island covered in shadowy vapours and steeped in all manner of deadly vibes. Before the game even starts, he's been emotionally traumatised by the weird emanations from the isle. However, despite having images of death breathed into his face, the intrepid Sir Art, persists and so the game commences.

The game is joystick controlled and there are various helpful items. The magic orbs come free at the start of the game and Sir Arthur has a great overarm swing for throwing them. They're quite useful for pulverising antagonistic brutes and the supply seems to be unlimited. The shovel is fairly easy to get hold of if you can outwit two ant-like beings and it can be used to dig up extra lives. I never laid eyes on the magic energy cloak but apparently it makes you virtually invincible.

A lot of the caverns look very similar and the screen with the fire-breathing dragon is incredibly boring because you need 10 direct hits with magic orbs before the dragon finally dematerialises.

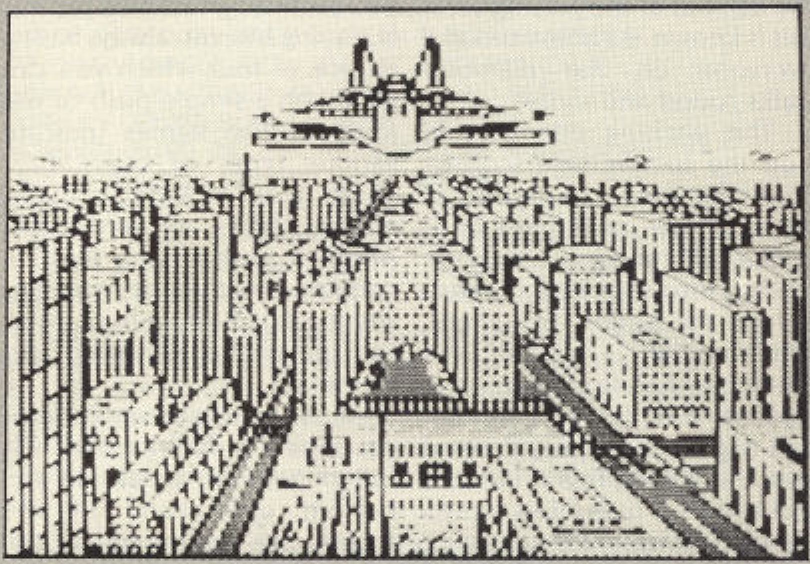
I think you'd need a lot of patience to get through it and feeding the budgie would probably be more entertaining.

M.C.

Alternate Reality

US Gold £19.95 — two disks, joystick' optional

Originality 7 Playability 9 Graphics 6 Value For Money 9



IT WAS A QUIET DAY IN THE city. Or at least it was until a spaceship arrived and kidnapped you. Things get curiouser and curiouser as you find yourself in a room with only one exit. On the outside is the City of Xebec's Demise. There seems no alternative but to go through the door.

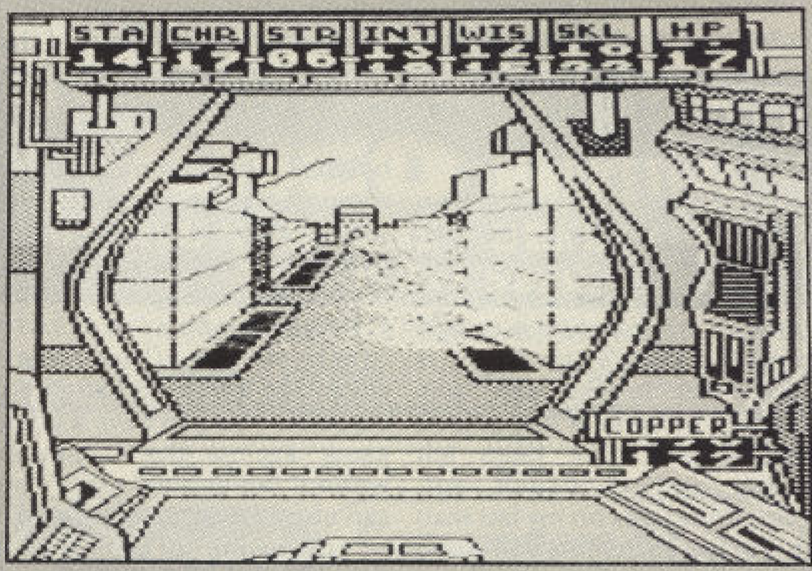
Alternate Reality is an extremely sophisticated fantasy role playing game. As soon as you step through the door, a set of spinning numbers above your head stop rotating and your characteristics in various fields are set. The higher they are, the better chance you have of surviving the many and assorted hazards to be found in the City. Stamina, charm, strength, intelligence, wisdom, skill and hit points all affect the outcome of assorted encounters which you will experience. You are also given a certain number of copper pieces with which to seek your fortune. Trying to pick a decent set of characteristics before you step out is nigh on impossible as the numbers are all scrolling at different speeds. Personally, I try to get as much money as possible.

You start off at the Floating Gate. This is the safest part of the city but incredibly expensive. 12,000 odd copper pieces seems a lot for a sweater when you only have 209. Movement is controlled by joystick or keyboard. You can move forwards and backwards, turn left and right. There is a small picture in the centre of the screen showing your current view. There are many buildings to enter where the proprietors are interested in relieving you of your money. Shops sell assorted items of clothing, mostly way beyond your current means. A compass proves to be a wise investment though. Smithies give you a chance to haggle over the price of various weapons. Banks invite you to open one of several types of account and will value gems and jewellery for you. There are inns and taverns, guilds and healers to be sought out and used.

One thing that makes Alternative Reality stand out from its competitors is its attention to detail. For example, when you go to an inn to spend the night, you must decide whether you want to sleep rough on the floor, take a room with no bath or shared bath and so on up through varying degrees of comfort to the best suite in the house.

It is not long before you encounter someone or something. Who, if anyone, has surprised whom is determined and you go into one of two menus. Engaged is basically offensive as you try to attack it. Disengaged allows you to try and charm or trick your opponent. Of course, you may just want to walk away – if you are allowed to. Your progress is monitored by gaining experience points and going up in skill levels. These are awarded for killing monsters and finding treasure etc.

Combat is decided according to a variety f factors: What weapon is being used – each one has an attack and defence value; whether you attack aggressively, normally or defensively; what your strength and skill levels are; how many hit points you have. When you hit or are hit, a number of hit points of damage are inflicted. When your hit points reach zero, you die. You may get poisoned or diseased in combat and have to visit a healer. You may of course also find some treasure.



Treasure comes in a variety of forms ranging from the simple monetary through potions and gems to weapons. Anything apart from money is potentially magic or cursed. Potions may be beneficial or poisonous, weapons may have special properties or be cursed. A cursed weapon cannot be dropped but a guild will remove a curse for a suitable fee.

The denizens of the city are many and various. There are muggers, zombies, hobbits, wizards, couriers, merchants, gladiators, noblemen and many, many more including the bane of my life – brown mold (I have never yet survived an encounter with it). There is also rumoured to be a Night Stalker that may or may not exist.

Time and weather play an important part in the game. Rain and darkness attracts the more unsavoury type of characters. You need to rest frequently and buy food and drink. The sun can be seen rising and setting and you need to visit inns to find out the time and date, especially if you don't want to be out after nightfall.

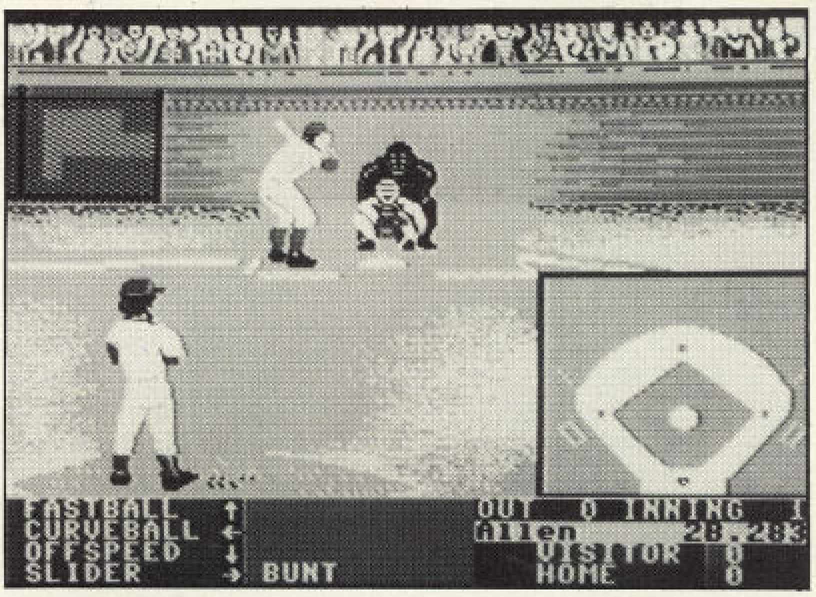
There are several areas of thegame currently inaccessible to you. This is because several expansion kits are planned – the Dungeon, the Arena, the Palace, the Wilderness, Revelation and Destiny. As you can gather, your quest of returning to Earth or taking revenge on your captors is not going to be accomplished in a single night and so you will need to save your current position to a blank disk fairly frewuently. The game comes nicely packaged with a booklet, an instruction leaflet and a partially completed map of the City.

It took me some time to get into Alternate Reality. As the game takes place in real time, I found that I had only just read the various options when they disappeared. I also found the frequent switching of disks frustrating, not through any fault of the game but because the C64's disk drive is so slow. However, slowly but surely, I began to get the feel of things and the satisfaction of tricking my first merchant, charming a gladiator and wrestling a zombie was considerable. The packaging claims that the game is the ultimate role-playing fantasy game. For once, I agree wholeheartedly.

G.R.H.

Hardball
US Gold £9.95

Originality 7 Playability 9 Graphics 9 Value For Money 9



IT IS THE NINTH AND LAST innings in the last match of the World Series (well, you can dream can't you?) and I am losing 4-3.

My first man takes an almighty swing and the ball goes soaring into the outfield. Unfortunately, there is a fielder underneath it and he is caught out. The team's star batter is next man in but can only manage three wild hits at balls that should have been left alone and is struck out. Everything rests on my last man in. He leaves the first ball alone but swings at the second – a fraction early and the ball goes straight to an infielder who has no trouble at all in throwing to first base and running me out by a mile. Determined to get revenge, I press the button for another game, oblivious of the fact that it is already well past two in the morning.

Hardball is a superb baseball simulation featuring some of the best graphics yet seen on the C64. Baseball is not particularly well known in this country but this game should stimulate considerable interest. Another huge advantage is that it is extremely playable. You can have a good game against the computer without being outclassed. You can also play against a friend.

For anyone not familiar with the game of baseball, it is based on the old British game of rounders. The batting side tries to hit the ball and move men round the four bases on a diamond. If a batter gets all the way round, he scores a point for his side. If he manages to hit the ball out of the playing area, this is known as a home run and everyone on the diamond walks round and scores.

The pitching side tries to stop the batting side scoring. Batters can be out in a variety of ways. He can be caught as in cricket. He can be run out if the fielders throw the ball to a base before the batter runs there. When a ball is pitched, the batter can decide to hit or leave it alone. If he leaves it and the ball is going to miss a defined area, he is OK. This is called a ball and four balls allow him to walk to first base. If it was going to hit the target area, it is called a strike. Three strikes and the batter is out. If the batter swings and misses, it is a strike regardless of whether or not the ball was on target.

A game consists of nine innings for each side. Each innings lasts until three men are out.

Hardball allows you to select your team and make various substitutions and changes. Each player has his name displayed together with his various playing statistics. Until you are fairly happy that you know what you are doing, I would strongly recommend that you stick with the originally selected areas.

When you are happy with your team, it's on to the game itself. You decide what type of ball your pitcher is going to throw. These are decided for you according to which pitcher is playing but you always have a choice of four which you can pick with a simple push of the joystick. The names indicate various types of curve ball. These may include fastballs, sinkers, sliders and screwballs. Another push of the joystick decides where you are going to aim the ball. Release the button and the pitcher winds up and throws.

If the batter leaves it alone or misses, a ball or strike is called and the details in the bottom right hand of the screen are amended. If the batter hits it, the scene switches to a picture of the outfield. The fielder nearest the ball flashes and you must manoeuvre him accordingly, trying to catch or field the ball. If you pick it up, maybe with a spectacular dive, you must then decide which base to throw to.

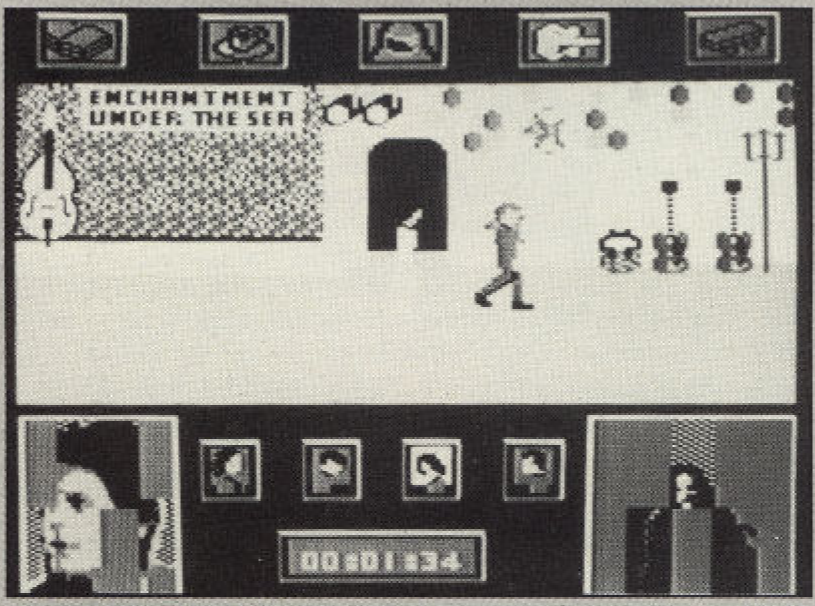
This continues until you get three batters out and then it is your turn to bat. A press of the button puts your man into batting stance. The ball is pitched and you have but a split second to decide whether or not to try and hit it. You may also get other choices such as whether to try and steal a base or bunt.

The animation and playability of Hardball really are superb. Beg, borrow, steal or even buy a copy!

G.R.H.

Back to the Future
Electric Dreams £9.95

Originality 6 Playability 6 Graphics 7 Value For Money 5



BASED ON THE FILM OF THE same name, Back to the Future sees our hero, Marty McFly going back in time to 1955. There he meets George and Lorraine his future parents. The problem is, Lorraine fancies Marty and he realises that unless he can persuade his parents to fall in love, the future will change irrevocably.

To get your parents to fall in love, they must spend a lot of time with each other. This you do by picking up an object in one of the four locations and using it to influence one of the characters in the game – as well as your parents, there is Biff the bully and Doc Brown. These objects – love poems, a guitar, an alien suit and a cup of coffee – cause the characters to stay still, follow you or turn away. You can move round rapidly by using a keyboard and can punch – and be punched by – Biff.

Your progress is determined by two photographs. As your parents stay together, so your picture builds up and this in turn results in the family picture getting completed. Do badly and the pictures start to disappear. On completion, a quick visit to Doc's house will take you Back to the Future.

The game's graphics are quite good, making use of icons for objects and characters. The game itself lacks substance and lasting appeal though. Not so much Back to the Future as back to the drawing board. **G.R.H.**

Sense of Adventure



Runescaster reveals the basic skills of adventuring and worms his way into some new software.

SEVERAL PEOPLE HAVE WRITTEN IN TO say they have been given adventures but do not know what to do with them! This may seem strange to regular readers of this column but do you remember how you got on with the first adventure you ever attempted?

Most of the games available today have quite good instructions, often including a list of words which will be recognised and usually a few general playing hints as well. Some tell you very little indeed. For those that are trying adventures for the first time, here are some basic rules for adventuring. Although these are intended for text input games, they apply equally well for any type of adventure game. Just remember that whether set in dragon infested dungeons or in a spaceship in the far future, all adventures are about puzzles – puzzles which you have to solve!

Rule number one is: Never get lost. This can obviously be somewhat of a problem, as adventures have many locations intended to make you lose your bearings – like mazes, one way passages or teleportation rooms. Many 'goodies' are also hidden in these out-of-the-way places.

Nevertheless, you must be able to draw a complete map of your travels even if it means you have to keep coming back to the same place and trying different moves.

The standard system is to draw a number of boxes, as in figure 1, joined by lines signifying the possible routes. Even if the program tells you the obvious exits from a location, get into the habit of trying all directions – programmers are devious creatures, there may be un-obvious exits!

Mazes are difficult to map. When you come across several locations that have the same or very similar descriptions – watch out! There are a number of variations on a similar theme, but all are begging the question how do you make each location unique, so that you can record its absolute position on your map?

The original system still works in most cases – this was to drop an item at each location (fig. 2), so that on returning there you would get a description that added the dropped item, so: 'YOU ARE IN A MAZE OF TWISTY PASSAGES – THERE IS AN AXE HERE'. With the last phrase supplying the unique definition.

As mentioned above, programmers are devious beggars and have produced a selection of ploys that are intended to stop you mapping a maze so easily! One of my favourites was when anything dropped sank into the muddy floor!

Often you find that there is one extra location above the number of objects you can carry (or are available). If you think about it, this also makes the last location unique – the only one without something there!

Just remember that there must be some way of solving this puzzle you just have to find it! 'EXAMINE' everything, perhaps the walls have distinctive markings or a mirror you are carrying reflects something different at one or

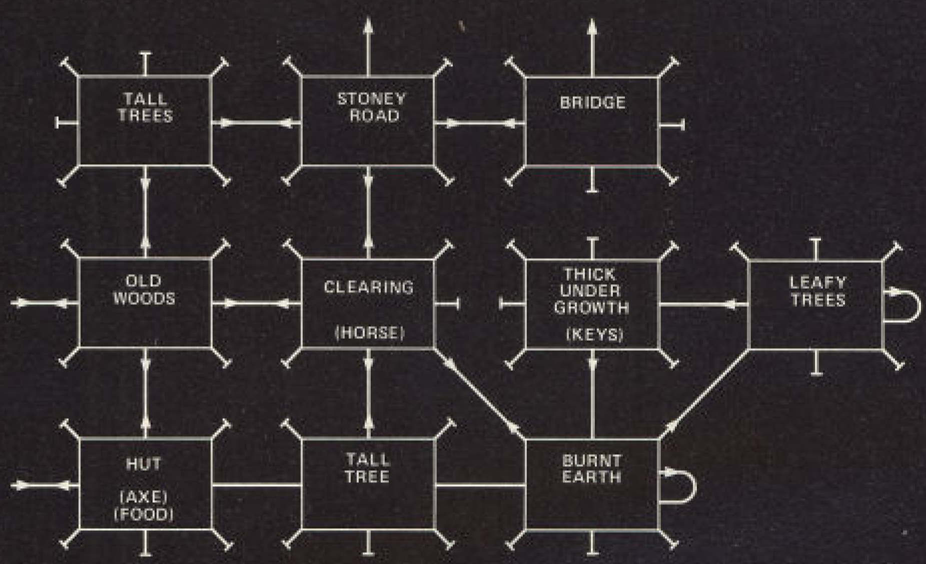


Figure 1: Standard adventure mapping

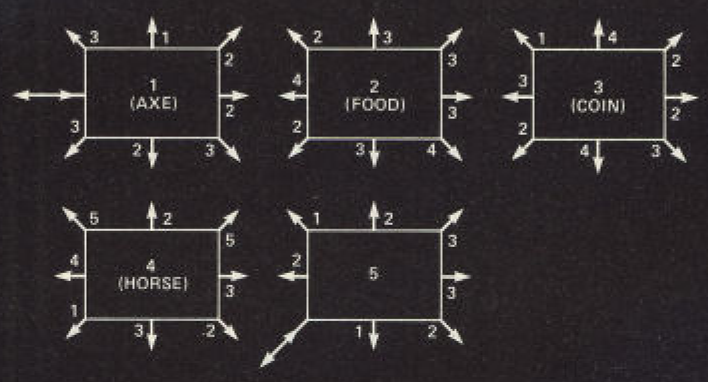


Figure 2: Maze mapping: Number locations and show where each movement will take you.

more locations! Look carefully at the way the description is written, perhaps there is some slight difference – an additional comma perhaps?

Rule number two is: Learn the language. Find out which words are understood and the way in which the input command has to be written. Some adventures will accept an input such as: 'GET ALL BUT THE RED BOOK AND RUN SOUTH AND SIT ON THE ELVEN KING'S THRONE'. Few experienced adventurers make use of such complex commands – certainly in the initial stages of learning what a game is all about.

It is much more usual to use the simple verb/noun type of input such as: 'GET SWORD' or if necessary 'RUN QUIETLY SOUTH'. But what you must do, is learn the words that will be understood – do you have to use 'EXAMINE', 'SEARCH' or 'LOOK'? Can you 'CLIMB' a bench or do you have to 'GET ON' it?

This familiarity with the program is important to get the most out of it – it does not take too long to learn a new games' idiosyncrasies but it is still time well spent.

Does the program support the use of 'HELP' and then give you the odd clue? It will almost certainly react to 'INVENTORY' and tell you what you are carrying. More importantly, does it allow you to 'SAVE' your present status and position, so that you can return in the future?

How do you repeat the location description after various messages have caused it to scroll off the top of the screen? Most will accept 'LOOK' for this command but some use 'REDEFINE'! There is a growing trend towards a standard 'Adventure Vocabulary' for all the normal requirements but watch out for the non-standard ones amongst them.

Do the commands have to be written in full (frustrating for slow typists!), or can words be shortened to the first three or four letters? 'CLIM MONO' is much simpler to type than 'CLIMB MONOLITH' and also helps to keep your concentration on the game rather than on the keyboard.

Rule number three is: Spot the pattern. Good adventures are like good crossword puzzles – the answers are obvious, after you've solved them! Also, like crosswords from different newspapers, they often seem to have their own individual patterns.

Do you have to 'EXAMINE' something twice before you find what you really want? Do you forever have to go forwards to find something you needed to solve an earlier puzzle? Are only objects made of wood significant – with those of metal inhibiting your progress?

Look for patterns or themes. Does this strange world which you have entered have its own laws? If so, you must learn them! A good adventure will have you sitting on the edge of your seat, sweating and intent on solving a problem that may mean life or death. To get the most out of it you must live the part and that means understanding its rules.

Finally, although one of the strongest attractions of an adventure game is the way in which the adventurer has freedom of movement and action, you must never forget the final objective. This may be to rescue the fair princess, save the world, capture a super-criminal, escape from an inhospitable planet or just find loads of treasure. But whatever it is, you will probably find more of a challenge in the main stream than wandering off on your own!

Christmas Bonanza

The month or so before Christmas saw a positive torrent of adventures appearing in the shops or promised by software houses. Far too many for us to deal with in any depth in one month! Some of these have been extensively advertised and hopefully will have given you some restful (?) moments between consuming all that turkey.

Followers of Level 9 games should already be deep into the latest in the Silicon Dream series *The Worm in Paradise*. That hard working team from Adventure International. Mike Woodroffe and Brian Howarth, have computerised the Fighting Fantasy book *Seas of Blood* by Steve Jackson and Ian Livingstone. Audiogenic has taken over the sales and distribution of *The Secret of St Brides* and at long last Melbourne House has released the first game in its *Lord of the Rings* series.

Ocean has a probable winner with its interpretation of the film *The Neverending Story*. Ariolasoft is distributing a novel, joystick driven, text adventure game – *Wild West* – where three input command options are displayed at each location. All you have to do is choose the right one! It may sound as though this should be an easy game to play but, when you work out the permutations, you'll see that it's not so.

US Gold has launched *Lucifer's Realm* with *Mission Asteroid* and *Masquerade* not far behind. In the last of these you play the part of a Mike Hammer type private-eye after a rather nasty Crime Boss – observation and deduction are your tools in trade. Use them well.

Duckworth has launched a new text only adventure – *The Odyssey* written by those adventurous brothers Gerrard. At least this latest epic from their key-worn fingers includes a 'SAVE' and 'LOAD' routine, so this time we won't have to keep our C64s running night and day to complete it!

The Worm Turns

Talk amongst adventure players sooner or later will get round to the games from Level 9. Early Level 9 games were text only but it was text filled with sparkling descriptions which made one wonder how it fitted into the computer's memory! As the demand for graphics grew, Level 9 met it with reasonable graphics (lots of screenfuls of them too) and still with lots of text to go with them.

For Level 9, things did not stand still, this software house looked for ways to improve not only the graphics but also the basic operating systems. Finally Pete Austin emerged from his soundproofed, darkened room with *The Worm in Paradise*.

The Worm is setting a standard that even Level 9 hopes will last for some time. It boasts the facility to recognise a vocabulary of over 1000 words. No longer will you see a word in a description and experience the computer's refusal to recognise its existence when you use it yourself. You may get the reply that the word is only part of the scenery but at least you know where you stand.

As in the last Level 9 offering – *Red Moon* – the game also features multi-tasking, so you can be typing in your next command whilst the pictures of your location are being 'drawn' to the screen. Those same commands can be pretty complex too. An example given is "EXAMINE ALL BUT THE HELMET, DUMMY AND LEOTARD AND GO EAST"!

This third, *Silicon Dream* adventure (the others were *Snowball* and *Return to Eden*) is a science fiction game set in the far future, 100 years after the events of *Return to Eden*. The scene is set on Eden, an apparent paradise of a planet. Robots are commonplace, humans work only 15 hours a week, no taxes are paid and everything in the garden seems lovely. Or is it? Surely an ideal place to live and penalties apparently only for evil-doers. Yours is the search to find out the underlying truth.

Unlike many adventures where movement is highly restricted by a series of puzzles, here there is plenty to explore right from the start. Type in 'SCORE' regularly to see if your actions have made any difference. One presumes that a higher score means a move along the path to understanding and deliverance.

Level 9 has always had a name for marvellous zany humour *The Worm* is no exception to this tradition. Tradition or not, this game can very well be read on more than one level and I wonder if some very basic 'Level 9 philosophy' is being dispensed in a very palatable manner.

Although you may wander around at will, there are plenty of opportunities for you to do things. As you might expect, some of these actions are not beneficial. The moral here, as always, is to 'SAVE' the game position fairly regularly.

If you don't receive *The Worm in Paradise* for Christmas, I would heartily recommend you get it as soon as possible

– there is plenty in it for novice and expert alike.

Swashbuckling For Everyone

The prolific team at Adventure International must share the ratings with Level 9 for producing this country's top adventures. Unlike Level 9, whose games are nearly all original ideas, A1 tends to specialise in taking existing books or characters (*Spiderman*, *Gremlins* etc.) and working magic to produce a version that is truly original.

Seas of Blood is based upon the Penguin Fighting Fantasy book of the same name. This is a multi-option adventure that has proved, together with others by the same authors, that this form of role-playing adventure book can and does, capture the imagination of the youngsters who buy them in their thousands. We now have to wait and see if the computer version can achieve the same popularity.

The story-line is fairly simple; you are the captain of the pirate vessel Banshee, to complete the game you must find and retrieve some 20 treasures and take them to the top of a mountain at the southern end of the Inland sea. Quite why you must do this is not clear but then no-one without imagination plays adventure games.

Not having 'played' the book I cannot compare the two products, all I can say is that the computer version is on a par with other AI recent releases. It has good colourful graphics and a similar operating system to the other games.

Input commands are given in the form verb/noun and the vocabulary understood is not extensive, in fact this facet of AI's programs is quite unsophisticated. Most of the time, if a word is not recognised, it is ignored! Strangely enough, although this can sometimes be slightly misleading, it does encourage a simple direct approach which is perhaps one of its attractions.

Note that the game is the first of the Fighting Fantasy titles – with the accent on fighting! Whenever you attack or are attacked, the screen displays a pair of spinning dice and a series of throws determines the outcome of the battle. Random factors such as these can ruin an otherwise good adventure game but this system works quite well.

Until you have found what has to be fought, the fighting aspect can make the game fairly short! Persevere and things become clearer. Some risks can be balanced out by judicious SAVEing of the game.

The game starts with you aboard the Banshee. To move you must type in 'SAIL SOUTH' etc., a bit long winded but that's the system. There are over 400 apparent locations but, as a very large number of these are the same view of the sea from the ship, don't get too excited! I also suspect that after a certain southern point a series of seascapes is repeated ad-infinitum.

This sea view can pose certain problems – like mapping. It is very important to know where you have been, or more to the point, where you have not been! You must also learn where you can find provisions, for without them your crew will mutiny and the game will end.

Each sailing move costs you one unit of these provisions and this can indirectly help you with mapping – check your number of sailed moves by noting the provisions left.

On a couple of occasions I ended up stranded on the shore, unable to 'BOARD BANSHEE' or 'GO BANSHEE'. Whether this is a bug or merely a game puzzle, time will tell. Note the use of 'GO XXXX'. AI often uses this syntax, so if you find you cannot get to what is obviously an interesting location, try 'GO ENTRANCE', 'GO PASSAGE', 'GO HOLE' etc.

All the sea scenes pose an interesting problem in accurate mapping. *Seas of Blood* has a certain novelty and encourages you to continue if only to check your own stamina! The puzzles are interesting without being particularly brain-bending and followers of Adventure International should enjoy this one.

Tape Of The Film

Ocean Software seems to have discovered a winning formula for an adventure game at the first try and it's nice to see some original thinking producing an interesting new approach. *The Neverending Story* comes as four parts on two cassettes, each loading pretty quickly in about two and a half minutes.

The game follows the general story-line of the film, with the powers of evil relentlessly attacking the land of Fantasia. This 'All-consuming Nothing' can only be halted and reversed by someone outside the reality of Fantasia.

Somewhere on our world is a little boy who finds a book telling the story of Fantasia. He begins to believe, and enters and travels about this other world in the form of a local lad called Atreyu.

The graphics are colourful and novel. One main scene is displayed across the top of the screen, this is then overlayed with a small picture showing either who is present or where you are. A further six small pictures show what you have with you and could be a little difficult to decipher.

The command interpreter is simple-minded, with only a very small vocabulary, requiring only the simplest verb/noun input. This is the weakest part of the game. You have to 'GET FALKOR' (the flying Luckdragon) before you can 'FLY WEST' etc. and having finished with his services you then have to 'DROP FALKOR'! Neither does it recognise the 'adventures friend' 'EXAMINE'.

The game could make a good primer for the novice adventurer since there are plenty of places to explore right from the start. The puzzles set are not terribly difficult but will often require a little thought! But there is one snag – no 'SAVE' and 'RESTORE' facility.

The first part has about 40 locations and although you die if you stay too long in the swamps there is a fairly logical selection of 'goodies' to be found and used. Most things have a purpose even if they are not necessary to complete the game.

Mapping this part is not difficult and will save a lot of wandering around, so do it! Generally the text will give you the right hints at the right time. If in doubt remember that 'fire destroys'.

Once you have loaded the second part – get ready for the frustrating bit. There are over 25 death traps within six moves of the start! If you take a wrong step you have to re-load part one! This is where the lack of a 'SAVE' game position becomes something of a classic error in programming. Other computer versions of the game have a 'SAVE'!

I do not think it will spoil the fun if I advise you not to travel west or south of where you start. Neither should you travel more than two moves to the east until you have moved north twice. You can still die by going too far north but read the text and take care. There are still nine lethal moves!

Your wonderful dragon steed disappears at the start of part two but reappears in the nick of time when you have discovered what you need to know. So persevere.

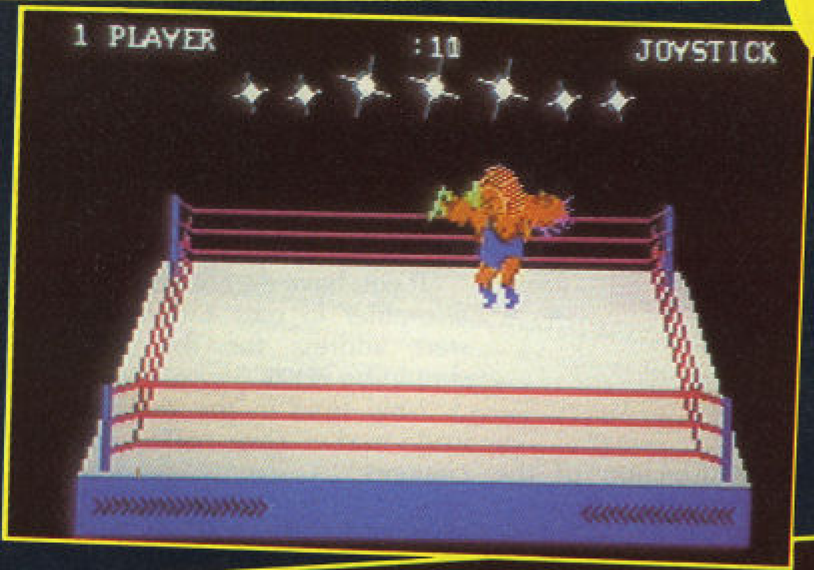
Part three has no lethal traps and is fairly easy providing you don't panic when faced with a maze. These come in three varieties – wood, stone and glass! Finally you will meet the Empress and realise that you have given Fantasia the chance of a future after all.

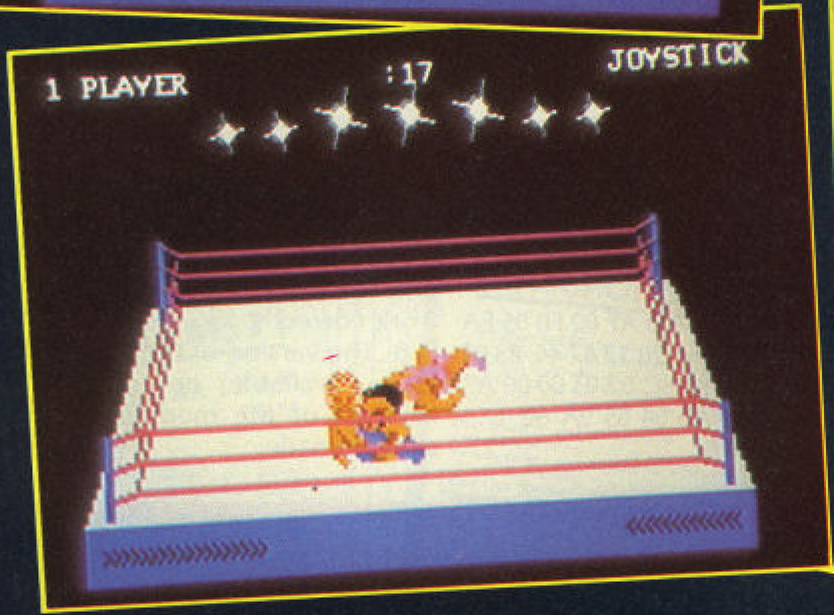
There are other characters who appear from time to time but they do not have any independent part to play in the action and merely add atmosphere.

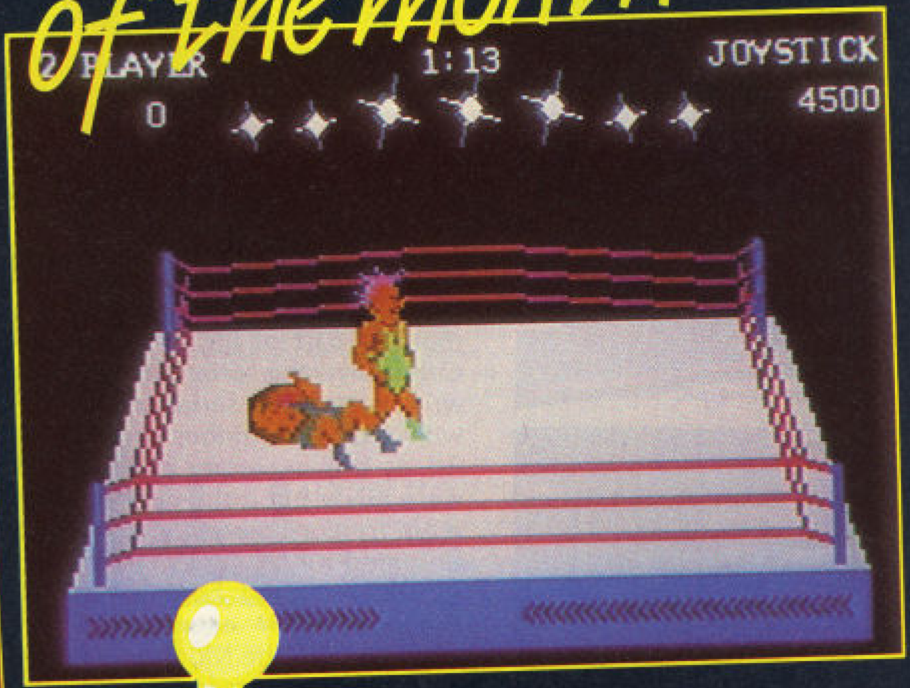
Make a note of the things you find and try and find a use for them. Never be without the good-luck charm, Auryn, for longer than necessity dictates! Even with its faults, an enjoyable adventure.

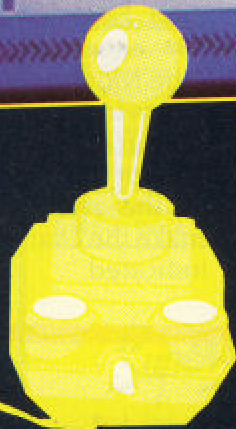
Get yourself into a half nelson with Rock'n'Wrestle from Melbourne House.

GAME of the month









YOU COULD PROBABLY COUNT ON the fingers of one hand the computer programs that would appeal to your granny. Well, you can now add one more to the list, Rock'n'Wrestle from Melbourne House. Load up this and it's just like any Saturday afternoon spent watching Bad Barney taking on Gorgeous Greg in the wrestling ring.

Not only is this a great game to play but it is great fun to sit back and watch. Play is probably a slightly tame word to describe the complete battering that you are bound to suffer as soon as you take control of your joystick. My advice is not to bother playing the game at all but rather sit back and be entertained.

The program has an excellent demo mode. You can sit back in the comfort of your own armchair and watch each of the 10 characters challenge each other in the ring. You can hear them scream as they are dropped on their heads. Watch the ring bounce as their opponent lands on top of them and hear the ref count them out.

The action can only be described as superb, obviously the graphics do lack a little realism but they aren't as bad as you might expect, the wrestlers perform all of the well-known moves, full nelson, back breaker, atomic drop and many more. It's quite good fun to watch one of the wrestlers being swung around in circles and then let go into one of the ropes, which give way very realistically as he hits them. Meanwhile the other player is getting ready to knock him to the floor as he bounces off, all accompanied by the necessary grunts and wheezes that you would expect in a real wrestling game.

No doubt there are a few wrestling and games fanatics who can't wait to get in the ring, personally I feel safer just watching. If you really must get into the ring and have a chance to look at the colour of the canvas then there are both one and two player options, the characters being controlled by either joystick or keyboard. The two player game is an excellent way to solve family arguments.

Controlling your players is extremely difficult as there are more than 35 different manoeuvres that you can make with the joystick. To make things even worse the direction that you must push the joystick in order to perform a specific function depends on which direction the character is facing it gets very confusing.

A typical move would be to approach the opponent from the front, push the joystick in the direction that the character is facing and then towards his back, this will grab your opponent. Next you should keep the joystick held back so that you can lift your opponent into the air. Now you can aeroplane spin him by jiggling your joystick left and right. Then by pulling the joystick back you can drop him unceremoniously with a pile driver.

At least, that's what's supposed to happen, in my case it was usually my player that ended up being dropped on his head only to be crushed by the other player and counted out by the ref.

Control sounds complicated, and it is. Don't worry though, if you are willing to spend a couple of house playing around you'll soon grasp the controls and then you can start to get your own back on your opponents.

There are 10 combatants including yourself. Each character is extremely well displayed and has varying characteristics.

Lord Toff is the most dangerous character of all, he plays very carefully and knows all the moves. Masked Bad Barney really likes to hurt his opponents, Flying Eagle is extremely agile, how he manages to keep his headress on I don't know. Next comes Vicious Vivian the local purple haired punk, and on they go.

The first charact er you'll find yourself taking on as Gorgeous Greg is Redneck McCoy. Take it from me, he really likes to drop you to the floor with an atomic drop, you don't get up very often from one of these. Perhaps I'm just chicken but I did find it easier to run and kick him every so often rather than trying to take him on, I usually ended up the worse for it if I did.

Rock'n'Wrestle is certainly one of the best games that I have come across. Not only is it great fun to play but you could keep the whole family entertained by simply letting them watch it, that's very unusual for a game.

If you own a C64 then you really must add this to your collection, you won't regret it.

Touchline
Program – Rock'n'Wrestle
Distributor – Melbourne House
Price – £9.95, £14.95 disk

MACH FIVE

Steve Carrie corrects a slight printer problem.

A NUMBER OF PEOPLE SEEM to be having a problem with the TTY command in the monitor part of the program. Originally, the program was written to use a serial printer with a device number of two. All of the necessary changes were made but it still does not seem to work correctly with a Commodore printer.

You will find here a few changes which need to be made to both the Monitor and the editor program in order for them to work correctly. You can use the monitor to change both itself and the assembler but you must make the changes in the order indicated.

Monitor

Use the M function of the monitor to alter the following:

```
8891 should be 4
889D should be 4
88A2 should be 4

88A4 should be 0
```

next enter the following with the M instruction

```
8E80 20 23 87 AD AF 02 F0 05 EA
EA 20 A8 8E 60 20 12 87 4C 83 8E
8EA0 20 A8 8E 6C 02 03 00 00 A2
04 20 C9 ff A9 04 85 9A 60
89E2 20 80 8E
8A4A 20 8E 8E
88B5 4C A0 8E
887E EA EA 4c A8 8E
```

now use the save function of the monitor to save the program with a

START ADDRESS 8200
END ADDRESS 8F00

If you have entered the dis-assembler in part 4 then the start address for the Save should be 7D00.

Now load in the monitor and make the following changes:

```
9CAA EA EA 4C A8 8E
9D46 EA EA 4C A8 8E
9DAA EA EA 4C A8 8E
```

Now save the assembler with:

START ADDRESS 9000
END ADDRESS 9DC0

The TTY command should now work correctly.
N.B. The version of the MACH series available on cassette have all of the modifications already made.

LANGUAGE LAB— PROLOG

David Janda becomes bilingual once again to bring you an analysis of PROLOG.

ALL THE PROGRAMMING LANGUAGES covered in this series so far have one thing in common; they are designed to be used for a specific types of task: that is, the language is best suited when used to solve a particular type of problem. One such example is COBOL (COmmon Business Orientated Language). COBOL is used for data processing in the business environment. Another example is FORTRAN (FORmula TRANSlation), which is used by the scientific community for number-crunching tasks.

These and other languages rely on the programmer being aware of what data he is going to work on, and in what format the data is. This can take up a lot of the programmer's time, which in a sense is wasted, because the programmer should be dealing with the task at hand – solving the problem!

The users are no better off either. There are very few programming languages which allow the user to ask 'what if?' type of questions – thus restricting them. However all is not lost – there is PROLOG.

PROLOG (PROgramming in LOGic) is fast becoming one of the most popular programming languages. Those of you who are aware of Artificial Intelligence (AI) will no doubt have heard about PROLOG, as it is a very popular language in this field. So much so that the Japanese have decided to adopt PROLOG as the primary language in their fifth generation computer project.

But what makes PROLOG programming different from programming with 'conventional' languages?

Imperative And Declarative

Programming as we know it requires the programmer to tell the computer exactly what to do by means of a set of instructions. The computer acts upon these instructions 'blindly' and does not deviate from them. This is also applicable to so-called decisions making features such as the IF construct – you are still telling the computer what to do. This is the imperative style of programming.

It is very strict and not very flexible. It would be far better if it were possible to describe the problem in general terms by describing the problem in an English type manner. That is what we do every day when we describe a problem. Now, providing the description of the problem contains all the data necessary to solve the problem, it should (in theory) be possible for a computer to find the answer!

To actually do this, the computer would need instructions telling it how to use the information that it is given. This is the declarative style of programming.

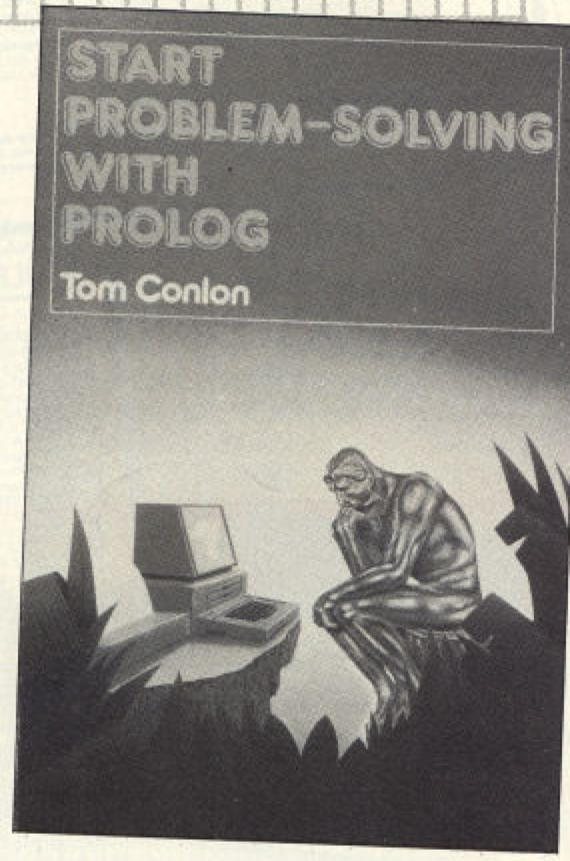
Describing the problem should not too different from describing a problem in real life e.g:

The pain is on the right side of my chest

and

My head spins if I climb the stairs

A declarative type of program requires a description which takes the form of a set of facts and rules, and the problem above includes these.



FACT: The pain is on the right side of my chest
RULE: My head spins if I climb the stairs

Here are some facts and rules from a declarative program
1 Asthma sufferers should avoid smoky atmospheres.
2 The atmosphere in Piccadilly Circus is smoky.
3 Henry is an asthma sufferer.

The problem is:

What should Henry avoid?

The program should answer: Piccadilly Circus.

Now, it cannot be said that PROLOG can be used to solve problems in such a free-hand style as described above, but it can be used in a similar manner.

The Language

A PROLOG program consists of a set of rules and facts which make up a program description – a program or database. Both facts and rules are referred to as sentences which must be written in a precise form known as a sentence form. Any questions put to the program are called queries.

PROLOG itself (the core language) is a little hard to understand and learn. Because of this, Logic Programming Associates (LPA) have included a user-friendly 'front-end' which makes PROLOG easier to use for beginners. All the examples are in SIMPLE. The following examples are extracts from the thoroughly excellent PROLOG tutorial, supplied with the package – *Problem Solving with PROLOG.*

Imagine a party. We wish to make various observations about what is going on. Bill is sitting gazing at Jean, and it's obvious that he likes her. Describing this fact as a PROLOG sentence is fairly straight forward

likes (Bill Jean)

Gossip is rife, and we discover that various people like each other.

likes (Diane Colin)
likes (Janet Ian)

The structure for PROLOG facts is as follows. First there is the predicate 'likes' which is followed by a bracketed list of terms called arguments. Predicates roughly correspond to verbs in English, and arguments to nouns. Notice that like English, even though Diane likes Colin, it does not mean that Colin likes Diane!

We would also observe that

enjoys (Diane rock)
enjoys (Jean reggae)

In this example the predicate is 'enjoys' and the two arguments are the name of the individual followed by the type of music.

A list of arguments is not restricted to two. Sam is acting barman, and is serving drinks to the guests:

gives (Sam Diane gin-and-tonic)
gives (Sam Colin cola)

Two things are worth observing. First, the hyphens are used to ensure that 'gin and tonic' is treated as one argument. If spaces were used each word would be treated as a separate argument. It is also important to be consistant about the number of arguments and their positions. For example:

gives (Sam Colin cola)
gives (Sam Gin david)

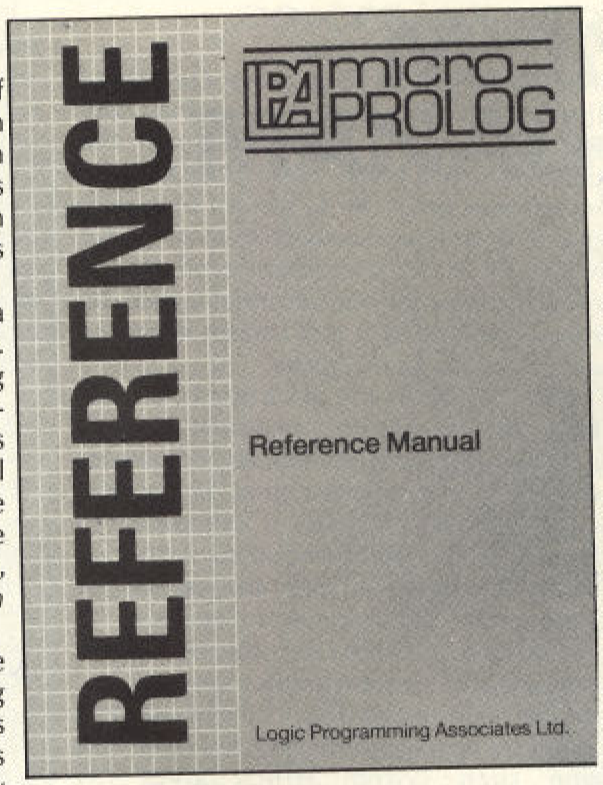


Table 1 — The Party Database

```
likes (Bill Jean)
likes (Diane Colin)
likes (Janet Ian)

enjoys (Diane rock)
enjoys (Jean reggae)
enjoys (Ian heavy-metal)

smokes (Diane)
smokes (Ian)

gives (Sam Diane gin-and-tonic)
gives (Sam Colin cola)
gives (Sam Jean white-wine)

ill (Janet) if smokes (Janet)

partners (Jean Ian) if enjoys (Ian heavy-
metal)
partners (Janet Ian) if
   enjoys (Ian heavy-metal) &
likes (Ian Bill)
partners (Diane Sam) if
   gives (Sam Diane gin-and-tonic) &
   not smokes (Sam)
```

PROLOG will not alert you to the fact that Sam gave the Gin a David!

Meanwhile back at the party, Janet has informed us that she becomes ill if she smokes. This is a rule. A PROLOG rule has two parts; the consequence and the condition

ill (Janet) if smokes (Janet)

The consequence 'Janet is ill' is true if the condition 'Janet smokes' is fulfilled. Each rule takes the form of an atom which describes the consequence part. This is followed by 'if' and atoms which make up the condition. If there is more then one condition 'and' and '&' are used to join them.

To demonstrate this, consider the situation where Diane will partner Sam if he gives her a gin and tonic and does not smoke. This would be written down as:

partners (Diane Sam) if
gives (Sam Diane gin-and-tonic) &
not smokes (Sam)

Asking Questions

With all these sentences of information (table 1) in the micro, it's about time we started asking some questions – or interrogated the database. Remember, even though the example is small, 'real-life' PROLOG programs could contain thousands of facts and rules.

A query can take the form of IS. To find out whether Janet likes Ian we would enter

is (likes (Janet Ian))

YES will be displayed. IS simply finds out whether a relationship is true or false. Another example is:

is (smokes (Bill))

NO will be displayed.

Now for the biggie! Does Jean partner Bill?

is (partners (Jean Bill))

YES is the answer! This time PROLOG had to do a little more work. PROLOG had to find out whether Bill likes heavy-metal.

Life is not so simple, and YES and NO are not very informative answers. We may wish to have the names of people as answers, and to do this WHICH queries are used. To find out who smokes at the party the following WHICH query can be entered

which (x : smokes (x))
Diane
Ian
No (more) answers

.. would be displayed. PROLOG is asking which term could replace x in the goal 'smokes(x)' in order to make the goal succeed.

The 'x' in the WHICH query is a variable, and they can be put to greater use because a variable can represent an unspecified term. For example, it is a known fact that someone is unhealthy if they smoke. To write such a rule in PROLOG the variable x would replace the 'someone'. The consequence part is:

unhealthy (x)

and the condition is

smokes (x)

So 'unhealthy' (x) if smokes (x)' can be added to the database. To actually find out who is unhealthy the WHICH query would be used in the form:

which (x: unhealthy (x))

PROLOG will display

Diane
Ian
No (more) answers

Summary

This article has just touched on the capabilities of the PROLOG language – there is much, much more!

The language is by no means simple to learn. The main reason for this is that the programmer has to adapt to declarative programming. Is it worth learning though? Well, PROLOG is already being used by many academic established dealing with AI. The fifth-generation of computers is being designed to be more friendly to use, and requires a language that will enable facts to be extracted with the minimum of fuss. PROLOG is one such language (LISP is another), and as stated already, the Japanese have decided to make it the official fifth-generation language for their new breed of super-computers.

6502 Micro-PROLOG

The Micro-PROLOG package consists of the system disk, reference manual and PROLOG tutorial guide. Although the package is a tutorial (the PROLOG is an interpreter only), there is a lot to it, with plenty of utilities and a lot to learn. LPA's Micro-PROLOG is an interpreted version of the language for interactive use on micros, and as such you cannot expect to write very large programs.

The heart of the package is a small built-in supervisor program which accepts programs, commands and queries written in a standard syntax. The supervisor can be tailored or extended by loading in programs (modules) written in PROLOG which can alter and extend the supervisor – a very powerful feature!

Besides the PROLOG interpreter itself, a number of these modules are supplied on the distribution disk. They include two program development systems; SIMPLE is one such system which gives the newcomer to PROLOG a user-friendly front end to the supervisor and is used throughout the tutorial manual. MITSI offers most of the features of SIMPLE, but provides an 'educational' type of front end with meaningful variable names and such like. Both these and any other executive extension take up RAM and are only intended to be used when learning the language.

Other programs on the disk include three trace programs (one for SIMPLE), an editor, a couple of error handlers and a package to facilitate the design of modules.

The biggest let-down with the package is the Reference Manual. It is written for use with other 6502 systems (BBC, Apple etc.), and is by far the worst manual I have ever read! The tutorial manual by Tom Conlon is excellent but written for those who are using SIMPLE.

Progressing from SIMPLE to the standard PROLOG syntax will be hard because of the badly written Reference Manual. Because of this I would recommend that anyone considering buying the package also considers buying a few books on PROLOG as well.

A couple of extensions (or primitives) for the C64 include IO for PEEKing and POKEing, POPEN for low-level stream opening and CAT for a disk catalogue. Graphics and sound are not supported with any extra primitives, but using IO it should be possible to use these features.

Loading micro-PROLOG takes under a minute and SIMPLE takes about a minute. 33844 bytes are free. 12% is taken up by the dictionary and the rest is 'shared' by the heap and stack and user programs.

Even though the user-manual is a let-down the Micro-PROLOG package is very comprehensive and well worth considering.

Commodore launches a program that is designed to please the bank manager.

MONEY MANAGEMENT



ARE YOU THE TYPE OF PERSON WHO dreads seeing a bank statement fall through your letter box? Do you never have enough money to pay your bills? Well, you no longer have an excuse. Commodore's Personal Money Management (PMM) program, for the C64, Plus/4 and C128 will come to your rescue.

Personal Money Management is designed to help you run two bank accounts. One of them should be a cheque account which should contain all money required for paying bills and general spending. The other is a deposit account which should contain any 'spare' cash that you have at any time.

Essentially PMM is a budgeting program. It requires you to decide exactly how much you spend on particular items each month. For example, if your gas bill is usually around £60 per quarter you know that you should budget for 20 per month. Provision is made for 10 different budgets and you can change these to suit your requirements. A couple of examples are given, one of a fairly well off family and the other of a poorly paid journalist. I must admit I do resent the journalists budget being for booze and betting, we're not all like that (honest!!). Each budget has a limit of £8000 per month (wishful thinking — Ed) though I'm sure that very few people will ever reach this limit.

Selecting the last budget account displays an extra menu. This allows you to enter all of your standing orders. The manual does recommend that you only put standing orders that are paid occasionally in this section and any monthly ones should be stored as one of the 10 budgets e.g. mortgage. This is because for each budget account you can view the last 14 payments that have been made. Since all of the standing orders are stored under one budget entry it is only possible to view the last 14 paid. When I tried to enter my own budgets for the month I found I soon ran out of entries. The only way to enter everything needed was to store items like monthly standing orders under one budget heading and enter a total cost as the payment. This causes no real problem but does mean that the program will only display the total of your payments rather than each payment made under this account heading.

Whenever you receive a bill, e.g. electricity, then these are entered. Obviously you need not enter any standing orders as these will already have been taken care of by the program. Again the program will only display records of the last 14 bills paid so if you want a check over the whole year it is better to calculate all bills for one month and input the total. Again not very friendly but it works.

One of the biggest problems that I found with the program is that you can't put your monthly salary into the computer. You have to enter what the computer calls 'extra income'. This is simply the difference between your budget account total and your actual income. For example if your budgets came to £250 per month and your earnings were £600 then you would have 'extra earnings' of £350. I thought that a computer was supposed to make things like this easier, surely it would have been possible for you to enter the actual income and let the computer work out the rest. Oh, I nearly forgot to mention that the program assumes that the 'extra income' will be going into the deposit account.

Once a Month

Once every month you must carry out the 'current month calculation' this is the time when the computer will tell you just how much you have or have not got. The computer gives the following information. The amount needed this month to pay all of your bills and standing orders. If you have spent less than your budgets then you are told to transfer the excess to your deposit account. This way you can get interest on the extra money.

If expenses are more than are in the cheque account you are informed just how much you need to transfer from the deposit account to your cheque account.

Last but by no means least is a list of the bills that you have said need to be paid, just in case you had forgotten.

Look at the Year

You can see at a glance just how well your accounts are doing at any time. You can check you current balances are working out, i.e. if you have spent more on a certain item than you had budgeted for. You can get a list of your expenses for the last 12 months and a record of the most recent transactions.

Verdict

If, like me, your one aim in life is to keep your bank manager happy and for some reason or other you never seem able to do so, then this is probably the program you have been waiting for. If you can set yourself up with reasonable budgets and stick to them then I can see no reason at all why the computer and your bank accounts shouldn't work together to make you a little better off.

As mentioned there are a couple of niggles or rough edges about this program, such as no wages entry, but these are soon overcome.

My advice is to keep you and your bank manager happy, go and buy a copy of PMM, it will no doubt pay for itself.

Touch line

Name — Personal Money Manager
Supplier — Commodore, 1 Hunters Road, Weldon, Corby, Northamptonshire, NN17 1QX. (0536) 205252
Price — £14.99

Keep a track of who's who and what's what with this C128 database from Frank Tout.

DATA BOS

IF YOU'VE EVER NEEDED TO store a lot of information, be it addresses or lists of books, then you probably used a good, old-fashioned card index box. This is great until you wish to look for some particular information. For example let's suppose that your card index box contains a list of names and addresses which have been filed by name. If you wanted a list of all the people who lived in London it could take you hours to look at each card to get the information you need. This is where Data Bos comes to the rescue.

Databos

```
10 fast:dim fi$(8,150),se(150):slow:goto5680
20 poke 53280,2:poke 53281,1:printchr$(147):printchr$(14)
30 color5,1:files$=""
40 scnclr:char1,16,1,"DATABOSS"
50 char1,3,3,"(1)Create File"
60 char1,22,3,"(2)Add New Record"
70 char1,3,5,"(3)View a Record"
80 char1,22,5,"(4)Tape Operations"
90 char1,3,7,"(5)Disk Operations"
100 char1,3,9,"(6)Print Record/Label"
110 char1,3,11,"(7)Modify Record/File"
120 char1,3,13,"(8)Search For a Record/Sort File"
130 char1,3,15,"(9)End Program"
140 ifadfile>0thencolor5,2:char1,5,20,"DATA IN STORE":color5,1
150 char1,3,23,"Please Type in a no. (1-9)"
160 get me$:me=val(me$):if me<1orme>9 then 160
170 if me=9 then 5640
180 rem
190 on me gosub 210,400,660,880,1010,2970,3750,4320
200 goto 40
210 rem
220 scnclr:char1,0,0,"Creating New Files"
230 char1,0,2,"How Many Fields Required? MAX. 8"
240 char1,0,4,"Space Bar To Abort"
250 get tfi$:if tfi$=""then 250
260 fast:if tfi$=" " then return
270 tfi=val(tfi$)
280 if tfi<1 or tfi>8 then 250
290 fi=tfi
300 rem*clear fields*
310 p$(1)="":p$(2)="":p$(3)="":p$(4)="":p$(5)="":p$(6)="":p$(7)="":p$(8)=""
320 slow
330 scnclr:char1,0,0,"Please Type in Fields,up to 9 Characters"
340 for a=1 to fi:print"field ";a;
350 input tp$
360 tp=len(tp$):if tp>9 then 350
370 p$(a)=tp$:tp$=""
380 next a
390 return
400 rem
410 rem*check file*
420 if fi>0 then 460
430 scnclr:color5,2:char1,0,0,"File Has Not Been Created":color5,1
440 for a=1 to 1000:nexta
450 return
460 adfile=adfile+1
470 if adfile>150 thenchar1,4,4,"No More Room":goto 600
480 scnclr:char1,7,2,"Record No.:-":printadfile
490 for a=1tofi:print"▒▒";p$(a)tab(11)chr$(58):next:goto 510
500 rem
510 print"▒▒▒▒▒";:fora=1tofi:fi$(a,adfile)=""
520 print"▒▒";tab(12);:input fi$(a,adfile)
530 if fi$(a,adfile)=""then fi$(a,adfile)="<"
540 rem
550 as=as+asc(fi$(a,adfile))
560 next a
570 rem*if all fields="<",delete record*
580 if as=fi*60 then adfile=adfile-1
590 as=0
600 printchr$(17)chr$(17);"to continue press any key"
610 printchr$(17)tab(2)"Space bar to return to menu"
620 get aq$:if aq$=""then 620
630 if aq$=" "then return
640 goto 460
650 return
660 rem
670 rem*ask for record no.*
680 input"▒which record no. ";wr$
690 wr=val(wr$):if wr<1 then 680
700 if wr=0 or wr>adfile then 760
710 rem*print record details*
720 print"▒▒▒▒▒▒▒record no.:-";wr
730 for a=1 to fi:atemp$=fi$(a,wr)
740 if atemp$="<" then atemp$=""
750 print"▒▒▒▒";p$(a)tab(11)chr$(58):print"▒▒▒▒▒▒";atemp$:next
760 print"▒▒▒▒▒press space bar to continue"
770 print"+ to scroll forward":print"- to scroll backward"
780 print:print"press any key to continue view"
790 get aq$:if aq$=""then 79
```

```
0
800 if aq$=" "then return
810 if aq$<>"+" and aq$<>"-" then 680
820 if aq$="+"and wr<(adfile)then wr=wr+1
830 if aq$="-"and wr>1 then wr=wr-1
840 if wr>adfile then wr=adfile
850 if adfile=0 then 790
860 goto 720
870 return
880 rem
890 printchr$(147);tab(4);"Tape operations"
900 char1,3,2,"(1)save/replace file"
910 char1,3,3,"(2)load file"
920 char1,3,4,"(3)return to main menu"
930 if adfile>0 thenchar1,10,20,"Data in Store"
940 char1,3,8,"Please type in a no.(1-3)"
950 get met$:met=val(met$):if met<1 or met>3 then 950
960 if met=3 then return
970 rem*get file name*
980 gosub 5550
990 on met gosub 1220,1550
1000 goto 890
1010 rem*******************
1020 rem*disk operations*
1030 rem*******************
1040 rem*menu*
1050 scnclr:char1,4,1,"Disk Operations"
1060 char1,3,3,"(1)Save/Replace File"
1070 char1,26,3,"(2)Load File"
1080 char1,3,4,"(3)Read Disk Directory"
1090 char1,3,5,"(4)Rename File on Disk"
1100 char1,3,6,"(5)Validate Disk"
1110 char1,3,7,"(6)Delete File on Disk"
1120 char1,3,8,"(7)Format/Rename Disk"
1130 char1,3,9,"(8)Return to Main Menu"
1140 if adfile>0 thenchar1,10,20,"DATA IN STORE"
1150 char1,5,22,"Please type in a no. (1-8)"
1160 get med$:med=val(med$):if med<1 or med>8 then 1160
1170 if med=8 then return
1180 if med<3 then gosub 5550
1190 on med gosub 1300,1630,1650,2560,2650,2710,2780
1200 goto 1050
1210 rem*write to disk/tape*
1220 rem
1230 rem*write to tape*
1240 scnclr:char1,1,1,"Insert Data tape":char1,2,6,"Press a Key when Ready"
1250 char1,4,8,"Space bar to abort"
1260 get aq$:ifaq$=""then 1260
1270 if aq$=" " then return
1280 open 1,1,1,file$
1290 gosub1370:goto40
1300 rem*write to disk*
1310 scnclr:char1,1,1,"Insert Data Disk into Drive"
1320 char1,2,6,"press any key when ready"
1330 char1,3,8,"space bar to abort"
1340 get aq$:if aq$=""then 1340
1350 if aq$=" "then return
1360 open 1,8,2,"@0:"+file$+",s,w"
1370 rem*write common to d&t*
1380 rem*write no. of fields,fi*
1390 print#1,fi;chr$(13);
1400 rem*write fields*
1410 for a=1 to fi:atemp$=p$(a)
1420 if atemp$="" then atemp$="<"
1430 print#1,atemp$;chr$(13);
1440 next a
1450 rem*write no. of records in file*
1460 print#1,adfile;chr$(13);
1470 rem*write data*
1480 for b=1 to adfile
1490 for a=1 to fi
1500 print#1,fi$(a,b);chr$(13);
1510 nexta,b
1520 close1:return
1530 rem*read from disk/tape*
1540 rem*read from tape*
```

DATA BOS

Data Bos is a database program for the C128. It is essentially an electronic card index box. Each card can contain a maximum of eight fields. For example if you wanted to file names and addresses you could use the following fields:

NAME
ADDRESS1
ADDRESS2
ADDRESS3
ADDRESS4
POSTCODE

Or, if you wanted to store a list of your book collection, you could use the following fields:

TITLE
AUTHOR
PUBLISHER
ISBN
PRICE

In fact the uses are limited only by your requirements.

So far we've done nothing that you couldn't do with a normal card index. However, Data Bos has two very powerful functions – search and sort.

Search will allow you to look through a file for a specific string; for example 'Smith'. You can search in two different ways. A Fixed search will find all the records (or cards) which contain the word Smith as the first item in a specified field. A Flexi Search will find all records in which the word Smith occurs anywhere in a specified field. Once the search is complete it is possible to get a printout of the records found.

Sort will allow you to store the records in a specified order. For example, you may wish to sort an add ress file by Name but, later you may want to sort it by Town. The Sort command makes this very easy.

Getting A List

It is possible to get a printout of records whenever required. Records can be printed out in two formats, a record list and label list.

A record list will print out the name of the file followed by the record number. Next comes a list of all the fields followed by their contents.

A label list will simply print out the contents of each record without the field name. This makes Data Bos ideal for storing a mailing list as you can print records directly on to labels. If you do wish to use labels, then make sure that when you set up the file you use enough records to fill a label, leaving any entries that are not needed blank. If you don't, then you may find that the addresses are not printed on the label correctly.

Saving It All

Data Bos has the facility to save all of your data to either cassette or disk. The disk option also has a replace function so you can save updated file. On cassette, to replace a file, you simply have to record over the existing one. However it is best to have two copies of everything just in case your tape gets twisted or you spill coffee on your disk. You don't want to type in all your data again.

Getting It All In

Data Bos is written entirely in Basic so you should have no problems entering it on your machine. The program is best entered in lowercase. Press the Commodore key and SHIFT to turn your computer into this mode.

DATA BOSS

```
1550 rem
1560 scnclr:char1,1,1,"Insert Data Tape":char1,2,6,"press a key when ready"
1570 char1,3,8,"space bar to abort"
1580 get aq$:ifaq$=""then 1580
1590 if aq$=" " then return
1600 open 1,1,0,file$
1610 goto 1770
1620 rem*read from disk*
1630 rem*read from disk*
1640 rem******************
1650 scnclr:char1,1,1,"Insert Data Disk into Drive"
1660 char1,2,6,"Press Any Key When Ready"
1670 char1,3,8,"Space Bar to Abort"
1680 get aq$:if aq$=""then 1680
1690 if aq$=" "then return
1700 if med=3 then 2010
1710 if med=4 then 2620
1720 if med=5 then 2670
1730 if med=6 then 2750
1740 if med=7 then 2910
1750 open 1,8,2,"0:"+file$+",s,r"
1760 rem*read common to d&t*
1770 rem*read common to d&t*
1780 a=0:fi=0
1790 rem*read no. of fields*
1800 input#1,fi
1810 rem*read in fields*
1820 a=a+1
1830 input#1,p$(a):if p$(a)="<" then p$(a)=""
1840 if st<>0 then close 1:return
1850 if a<fi then 1820
1860 rem*read in no. of records*
1870 input#1,adfile
1880 rem*read in data*
1890 b=0
1900 b=b+1
1910 a=0
1920 a=a+1
1930 input#1,fi$(a,b)
1940 if st<>0 then close1:return
1950 if a<fi then 1920
1960 if b<adfile then 1900
1970 close1:return
1980 rem******************
1990 rem*disk dir reader*
2000 rem******************
2010 open 1,8,2,"$,s,r"
2020 for l=1 to 71
2030 get#1,a$,a$
2040 if st<>0 then close1:print"disk error":goto 2420
2050 next l
2060 t$(1)="seq":t$(2)="prg":t$(3)="usr":t$(4)="rel"
2070 rem*get disk name*
2080 i=17:gosub 2480
2090 dname$=temp$
2100 rem*get disk id*
2110 i=2:gosub 2480
2120 did$=temp$
2130 rem*get op system*
2140 i=2:gosub 2480
2150 os$=temp$
2160 rem*get rest of block,not used*
2170 for l=1 to 44:get#1,a$,a$:nextl
2180 print"▣disk name:"dname$,"id:"did$,"os:"os$
2190 lt=0
2200 rem*get details of file*
2210 rem*get file type ,ty$*
2220 for a=1 to 8:get#1,ty$,a$,a$
2230 if st<>0 then close 1:goto 2410
2240 if ty$<>""then 2270
2250 for l=1 to 27:get#1,a$:next l
2260 goto 2370
2270 rem*get file name*
2280 i=15:gosub 2480
2290 n$=temp$
2300 rem*get file length*
2310 for l=1 to 9:get#1,a$:next
2320 get#1,l$,h$
2330 l=asc(l$+chr$(0))+256*asc(h$+chr$(0)):if l=0 then 2370
2340 lt=lt+l
2350 rem*print details*
2360 print n$tab(20)t$(asc(ty$)-128),l
2370 if a<8 then get#1,a$,a$
2380 next a
2390 rem*continue with dir*
2400 goto 2200
2410 print"▣no. of blocks free:";664-lt
2420 print"▣▣to read disk dir again press any key"
2430 print"▣space bar to return to menu"
2440 get aq$:if aq$="" then 2440
2450 if aq$=" " then return
2460 goto 2010
2470 rem*string constructor*
2480 temp$=""
2490 for b=0 to i
2500 get#1,b$:if b$<>chr$(160)then temp$=temp$+b$
2510 next b
2520 return
2530 rem**********
2540 rem*disk cmd*
2550 rem**********
2560 rem*rename file on disk*
2570 print"▣renaming file on disk"
2580 input"▣please type in old name";olnam$
2590 input"▣new name";nenam$
2600 a=len(nenam$):if a>16 then print"▣name too long!!":goto 2590
2610 goto 1650
2620 open 15,8,15:print#15,"r0:"+nenam$+"="+olnam$
2630 close 15
2640 return
2650 rem*validate disk*
2660 goto 1650
2670 print"▣validating disk"
2680 open 15,8,15:print#15,"v"
2690 close 15
2700 return
2710 rem*delete file on disk*
2720 print"▣deleting a file"
2730 input"▣name of file";dfile$
2740 goto 1650
2750 open 15,8,15:print#15,"s0:"+dfile$
2760 close 15
2770 return
2780 rem*format/rename disk*
2790 print"▣formatting(1) or renaming(2) disk ?"
2800 print"▣warning-above commands erases files on     disk"
2810 get q$:if q$=""then 2810
2820 q=val(q$):if q<1 or q>2 then 2810
2830 input"▣name of disk";dnam$
2840 dnl=len(dnam$):if dnl>16 then print"▣name too long!!":goto 2830
2850 if dnl<1 then print"▣name?!":goto2830
2860 if q=2 then 2900
2870 input"▣id of disk";did$
2880 didl=len(did$):if didl>2 then print"▣id too long!!":goto2870
2890 if didl<1 then print"▣id?!":goto2870
2900 goto 1650
2910 open 15,8,15
2920 if q=2 then 2950
2930 print#15,"n0:"+dnam$+","+did$
2940 close 15:return
2950 print#15,"n0:"+dnam$
2960 close 15:return
2970 rem**********************
2980 rem*print record/label*
2990 rem**********************
3000 if fi=0 then 430:rem if no files have been created
3010 rem*menu*
3020 scnclr:char1,1,1,"Printer Section"
3030 char1,3,3,"(1)print record selective"
3040 char1,3,4,"(2)print all records"
```

```
3050 char1,3,5,"(3)print lab
el selective"
3060 char1,3,6,"(4)print all
 labels"
3070 char1,3,7,"(5)return to
main menu"
3080 char1,10,10,"please typ
e in a no. (1-5)"
3090 char1,2,12,"■■ please
ensure that printer is on"
3100 char1,2,13,"and that pa
per is in printer■"
3110 get mep$:mep=val(mep$):
if mep<1 or mep>5 then 3110
3120 if mep=5 then return
3130 on mep gosub 3160,3260,
3350,3440
3140 goto 3020
3150 rem*print record select
ive*
3160 input"■print record no
.(0 to stop)";wr$
3170 wr=val(wr$):if wr<1 or
wr>adfile then return
3180 gosub 3560
3190 goto 3160
3200 open 4,4
3210 for w=1 to 2:print#4,"
":next
3220 print#4,chr$(14);spc(6)
"file name:";file$
3230 print#4,spc(6)"record n
o.";wr
3240 print#4,chr$(15)
3250 rem*print all records*
3260 print"■printing all re
cords"
3270 print"■press s to stop
"
3280 for wr=1 to adfile
3290 gosub 3560:rem go to pr
int routine
3300 get a$:if a$="s" then w
r=adfile
3310 next wr
3320 return:rem return to me
nu
3330 goto 3020
3340 rem*print label*
3350 input"■print record no
.(0 to stop)";wr$
3360 wr=val(wr$):if wr<1 or
wr>adfile then return
3370 print"■how many fields
 to print,1-";fi"?"
3380 print"■0 to stop"
3390 get aq$:if aq$="" then
3390
3400 b=val(aq$):if b<1 orb>f
i then return
3410 gosub 3690:rem go to pr
int routine
3420 goto 3350
3430 rem*print all labels*
3440 print"■printing all la
bels"
3450 print"■how many fields
 to print,1-";fi;"?"
3460 print"■0 to abort"
3470 get aq$:if aq$="" then
3470
3480 b=val(aq$):if b<1 or b>
fi then return
3490 print"■press s to stop
"
3500 for wr=1 to adfile
3510 gosub 3690:rem go to pr
int routine
3520 get a$:if a$="s"then wr
=adfile
3530 next wr
3540 return:rem return to me
nu
3550 rem*print record*
3560 open 4,4
3570 for w=1 to 2:print#4,"
":next
3580 print#4,chr$(14);spc(6)
"file name:";file$
3590 print#4,spc(6)"record n
o.";wr
3600 print#4,chr$(15)
3610 for w=1 to 2:print#4,"
":next
3620 for a=1 to fi:atemp$=fi
$(a,wr):b=len(p$(a))
3630 if atemp$="<" then atem
p$=""
3640 print#4,spc(3)p$(a)spc(
15-b)atemp$chr$(10)
3650 next
3660 for w=1 to 3:print#4,"
":next
3670 print#4:close4:return
3680 rem*print label routine
*
3690 open4,4:print#4," "
3700 for a=1 to b:atemp$=fi$
(a,wr):if atemp$="<"then ate
mp$=""
3710 print#4,spc(2)atemp$
3720 next a
3730 print#4," "
3740 print#4:close4:return
3750 rem**********************
**
3760 rem*modify/delete recor
d*
3770 rem**********************
**
3780 rem*menu*
3790 scnclr:char1,1,1,"Modif
y/Delete"
3800 char1,3,2,"(1)Modify re
cord"
3810 char1,3,3,"(2)Delete a
Record"
3820 char1,3,4,"(3)Delete al
l records"
3830 char1,3,5,"(4)Delete a
field"
3840 char1,3,6,"(5)Return to
main menu"
3850 char1,5,8,"Please type
in a no.(1-5)"
3860 get mem$:mem=val(mem$):
if mem<1 or mem>5 then 3860
3870 if mem=5 then return
3880 on mem gosub 3910,4090,
4200,4230
3890 goto 3790
3900 rem*modify record*
3910 print"■state record no
."
3920 print"■0 to return to
menu■":input wr$
3930 if wr$="0"then return
3940 wr=val(wr$):if wr<1 or
wr>adfile then 3910
3950 print"■■■■■■■■recor
d no.";wr
3960 for a=1tofi:atemp$=fi$(
a,wr)
3970 if atemp$="<"then atemp
$=""
3980 print"■■■";p$(a):pri
nt"■■■■■■■■■■■■■■"
;atemp$
3990 next
4000 for a=1tofi:print"■■■
":next a
4010 for a=1tofi:input"■■■
";fi$(a,wr)
4020 next
4030 print"■■■■■to continu
e press any key"
4040 print"■■■space bar t
o return to menu"
4050 get aq$:if aq$=""then 4
050
4060 if aq$=" "then return
4070 goto 3910
4080 rem*delete a record*
4090 input"■delete which re
cord(0 to abort)";wr$
4100 wr=val(wr$):if wr<1 or
wr>adfile then return
4110 rem*move up all records
 after wr$*
4120 if wr=adfile then 4170
4130 for b=wr to (adfile-1)
4140 for a=1 to fi
4150 fi$(a,b)=fi$(a,b+1)
4160 next a,b
4170 adfile=adfile-1
4180 goto 4090
4190 rem*delete all records*
4200 input"■delete all reco
rds? y/n";aq$:if aq$="n" the
n return
4210 adfile=0:return
4220 rem*delete field*
4230 input"■deleting field:
which record";wr$
4240 wr=val(wr$):if wr<1 or
wr>adfile then return
4250 for a=1 to fi:print"■■"
;a;p$(a):next a
4260 print"■field to delete
"
4270 print"■0 to abort"
4280 get aq$:if aq$=""then 4
280
4290 aq=val(aq$):if aq<1 or
aq>fi then return
4300 fi$(aq,wr)="<"
4310 goto 4230
4320 rem**********************
***********
4330 rem*search for a record
/sort file*
4340 rem**********************
```

DATA BOSS

DATA BOSS

```
***********
4350 if fi=0 then 430
4360 print"▒searching(1) or sorting(2)? 1/2"
4370 print"▒space bar to return to menu"
4380 get  aq$:if aq$=""then 4380
4390 if aq$=" "then return
4400 if aq$="2"then 4980
4410 if aq$<>"1"then 4380
4420 rem*ask for information*
4430 in$="":print"▒"
4440 for a=1tofi:print"▒";a;p$(a):next
4450 print"▒▒▒▒which do you know 1-";fi
4460 get aq$:if aq$=""then 4460
4470 aq=val(aq$):if aq<1 or aq>fi then 4460
4480 print"▒▒▒▒what is your information":input in$
4490 rem*get length of in$*
4500 inl=len(in$):b=0
4510 ifadfile=0thenprint"▒no records in memory!!":fora=1 to 1000:next:return
4520 print"▒fixed search(1) or flexi-search(2)?"
4530 get q$:if q$=""then 4530
4540 q=val(q$):if q<>1 and q<>2 then 4530
4550 print"▒▒▒searching through file":if q=2 then 5390
4560 rem*fixed search*
4570 for s=1 to adfile:a$=fi$(aq,s):a=1
4580 x$(1)=mid$(a$,a,1)
4590 x$(2)=mid$(in$,a,1)
4600 x1=asc(x$(1)+chr$(0))
4610 x2=asc(x$(2)+chr$(0))
4620 if x1<>x2 then 4700
4630 if x1=x2 then a=a+1
4640 rem*when finished comparing*
4650 rem*return*
4660 if a>inl then 4680
4670 goto 4580
4680 b=b+1
4690 se(b)=s
4700 next s
4710 print"▒there are";b;"records containing"
4720 print in$:print"▒":ifb=0then 4790
4730 for a=1 to b step 9:d=0
4740 for c=a to a+8:if c>150 then 4770
4750 printtab(d)se(c);
4760 d=d+4
4770 next c:print:next a:print
4780 print"▒which do you want to view?"
4790 print"0  to return to menu":print"s  to do search again"
4800 print"p to print list"
4810 input wr$:if wr$="0"then return
4820 if wr$="s"then 4430
4830 if wr$="p" then 4880
4840 gosub 690
4850 goto 4710
4860 return
4870 rem*list to printer*
4880 open 4,4
4890 for a=1 to b step 6
4900 for c=a to a+5:if c>150 then 4920
4910 print#4,se(c),
4920 next c:print#4,chr$(13):next a
4930 close4
4940 goto 4810
4950 rem******************************************
4960 rem*sort file alphabetical/numerical order*
4970 rem******************************************
4980 print"▒which field to use when sorting?1-";fi
4990 get aq$:ifaq$=""then4990
5000 aq=val(aq$):if aq<1 or aq>fi then 4990
5010 for c=1 to 2:y=1
5020 b=aq
5030 print"▒sorting file"
5040 rem*compare two records at a time*
5050 for s=1 to (adfile-y)
5060 a=1:r1=s:r2=s+1:ex=0
5070 rem*get character from specified field*
5080 a$(1)=fi$(b,r1)
5090 a$(2)=fi$(b,r2)
5100 n$(1)=mid$(a$(1),a,1)
5110 n$(2)=mid$(a$(2),a,1)
5120 n1=asc(n$(1)+chr$(0))
5130 n2=asc(n$(2)+chr$(0))
5140 if n1>n2 then 5240
5150 if n1<n2 then 5310
5160 rem*if characters  the same then*
5170 rem*get next ones       *
5180 a=a+1
5190 rem*if a>field length , next set*
5200 rem*of records       *
5210 if a>24 then 5310
5220 goto 5100
5230 rem*swap records round*
5240 ex=1
5250 for x=1 to fi
5260 atemp$=fi$(x,r1)
5270 fi$(x,r1)=fi$(x,r2)
5280 fi$(x,r2)=atemp$
5290 next x
5300 rem*continue with sort*
5310 next s
5320 y=y+1:if y>adfile and ex=1 then y=1
5330 if y>adfile then return
5340 rem*do sort run again only*
5350 rem*if ex=1 see 14710      *
5360 if ex=1 then 5050
5370 next c
5380 return
5390 rem**************
5400 rem*flexi-search*
5410 rem**************
5420 c=0
5430 for s=1 to adfile:a$=fi$(aq,s):a=1:fl=len(a$)
5440 b=1:if a>fl then 5530
5450 x$(1)=mid$(a$,a,1):x$(2)=mid$(in$,b,1)
5460 x1=asc(x$(1)+chr$(0)):x2=asc(x$(2)+chr$(0))
5470 if x1<>x2 then a=a+1:goto 5440
5480 a=a+1:b=b+1
5490 if b>inl then 5520
5500 if a>fl then 5530
5510 goto 5450
5520 c=c+1:se(c)=s
5530 next s
5540 b=c:goto 4710
5550 rem***************************
5560 rem* file name for read/write*
5570 rem***************************
5580 print"▒please type in name of file":print"▒(up to 16 characters long)"
5590 input "▒";a$
5600 if a$="" or len(a$)>16 then 5580
5610 file$=a$:return
5620 return
5630 rem*end prog*
5640 graphic1,1:color4,5:color1,2
5650 scnclr:char1,12,12,"are you sure?y/n":circle1,160,100,80,40,,,,45
5660 get aq$:if aq$="n" then graphic0,1:goto40
5670 if aq$<>"y"then 5660
5680 graphic1,1:color0,12:color4,1:color1,16
5690 circle1,162,96,5,10
5700 fort=0to76step2:circle1,162,96,10+t,20+t,,,,44+t:nextt:paint1,161,111
5710 color1,8:char1,16,21,"databoss"
5720 color1,4:char1,2,23,"by frank tout 1985 space to continue"
5730 geta$:ifa$<>" "then5730
5740 graphic0,1:color0,12:goto20
```

Design a new character set with this editor for the C16 and Plus/4 by Nick Hampshire.

CHANGE YOUR CHARACTER

THE C-16 AND PLUS/4 COMputers, like the C64 and Vic 20 before them, have the capability of displaying a user defined character set.

The great value of user defined characters lies in the ease with which they can be used to create quite complex graphics displays. This feature is particular important in games programming. Most graphics games displays involve a considerable degree of repetition of a small range of patterns or shapes. There is also a common requirement in games programs to rapidly change or move areas of the display.

Normally, this would be carried out on a C64 using sprites, but the C-16 and Plus/4 do not have hardware sprite capability. Thus programmable characters are the best alternative way of achieving sprite-like displays.

Besides games, user defined characters are useful in any application which requires a special character set. Such applications include foreign languages and sprites, or scientific notation.

The problem with using the user defined character set on the C-16 and Plus/4, is that it requires access to the graphics control chip – TED. The TED chip is complex and unfortunately Commodore has published very little information on this device. It took a considerable amount of trial and error investigation to find the right locations and determine how they function (a complete list of TED locations and their functions will be given next month).

Having discovered the correct TED locations to create user defined characters, an editor program is required to simplify creation of the characters. The character editor included here is very versatile and comprehensive and can handle all the display modes of the machines.

It is very easy to use, and its commands are self documented. Each character is created using a magnified plot of the character. A normal size character is also shown, essential for getting the colour right in multi-colour mode. Once created the user defined characters can be saved to tape or disk for later and merging into a program.

```
1000 REM ******CHARACTER EDITOR******
1010 REM FOR MULTI COLOUR OR STANDARD
1020 REM CHARACTERS
1030 REM ****************************
1040 TRAP 2950
1050 POKE 1176,44:REM ENABLE CHAR ROM PEEK
1060 DEF FNS(Z)=3072+(R+1)*40+C+1
1070 DEF FNS(Z)=FNS(Z)-1024
1080 CH=0:TED=56280:SC=1:SL=4:BC=9:BL=4:CC=1:CL=7
1090 DIM P2(7):FORI=0TO7:P2(I)=2↑I:NEXT
1100 REM ****************************
1110 REM         PROTECT MEMORY
1120 REM ****************************
1130 POKE56,56:POKE54,56:POKE52,56
1140 POKE51,0:POKE53,0:POKE55,0
1150 ZZ$="":FORI=1TO6:READA:ZZ$=ZZ$+CHR$(A):DIRECTORYCHR$(A):NEXT
1160 ZZ$=ZZ$+"[illegible]+-MONSQ12!"+CHR$(34)
1170 CD$="[illegible]"
1180 GOSUB2620:REM PUT IN CHARS
1190 POKETED+18,PEEK(TED+18)AND251:POKETED+19,(PEEK(TED+19)AND3)OR56
1200 PRINT"[illegible]":COLOUR0,2,3:COLOUR4,10,4
1210 GOSUB2100:REM DO YOU WANT TO LOAD
1220 GOSUB1780
1230 REM ****************************
1240 REM MAIN INPUT LOOP FOR COMMANDS
1250 REM ****************************
1260 DO
1270 POKEFNC(0),0
1280 DO:DIRECTORYA$:LOOP WHILE A$=""
1290 POKEFNC(0),1+16*7
1300 LOOP DIRECTORY INSTR(ZZ$,A$)
1310 IFA$="+"THEN1560
1320 IFA$="-"THEN1630
1330 IFA$="▒"THENR=R+1:IFR=8THENR=0:GOTO1260
1340 IFA$="┐"THENR=R-1:IFR=-1THENR=7:GOTO1260
1350 IFA$="▮"THENC=C+1:IFC=8THENC=0:GOTO1260
1360 IFA$="▮"THENC=C-1:IFC=-1THENC=7:GOTO1260
1370 IFA$="N"THENCH=(CH+1)AND63:GOSUB2030:GOTO1260
1380 IFA$="Q"THEN2250
1390 IFA$="M"THENPOKETED+7,PEEK(TED+7)OR16:GOTO1260
1400 IFA$="O"THENPOKETED+7,PEEK(TED+7)AND239:GOTO1260
1410 IFA$="S"THEN1700:REM SELECT
1420 IFA$=CHR$(133)THENSC=(SC+1)AND15:POKETED+21,SC+SL*16:GOTO1260
1430 IFA$=CHR$(137)THENSL=(SL+1)AND7:POKETED+21,SC+SL*16:GOTO1260
1440 IFA$=CHR$(134)THENBC=(BC+1)AND15:POKETED+25,BC+BL*16:GOTO1260
1450 IFA$=CHR$(138)THENBL=(BL+1)AND7:POKETED+25,BC+BL*16:GOTO1260
1460 IFA$=CHR$(135)THENCC=(CC+1)AND15:GOTO2720:REMCHANGE CHARACTER COLOURS
1470 IFA$=CHR$(139)THENCL=(CL+1)AND7:GOTO2720:REM CHANGE CHARACTER COLOURS
1480 IFA$="1"THENZ1=(Z1+1)AND15:POKETED+22,Z1+Z2*16:GOTO1260
1490 IFA$="!"THENZ2=(Z2+1)AND7:POKETED+22,Z1+Z2*16:GOTO1260
1500 IFA$="2"THENZ3=(Z3+1)AND15:POKETED+23,Z3+Z4*16:GOTO1260
```

```
1510 IFA$=CHR$(34)THENZ4=(Z4+1)AND7:POKETED+23,Z3+Z4*16:GOTO1260
1520 IFA$="□"THENFORI=0TO7:POKE14848+CH*8+I,0:NEXT:GOSUB2030:A$="■"
1530 IFA$="■"THENR=0:C=0:GOTO1260
1540 GOTO1260
1550 REM ****************************
1560 REM ADD A POINT TO THE CHARACTER
1570 REM ****************************
1580 AD=14848+R+8*CH
1590 POKEAD,PEEK(AD)ORP2(7-(CAND7))
1600 POKEFNS(0),62
1610 A$="■":GOTO1350
1620 REM ****************************
1630 REM DELETE A POINT FROM THE CHAR
1640 REM ****************************
1650 AD=14848+R+8*CH
1660 POKEAD,PEEK(AD)AND(255-P2(7-(CAND7)))
1670 POKEFNS(0),46
1680 A$="■":GOTO1350
1690 REM ****************************
1700 REM SELECT A CHAR TO DISPLAY
1710 REM ****************************
1720 DO:OPEN1,0
1730 PRINTCD$;"CHAR NUMBER (0-63) ";:INPUT#1,CH$:CLOSE1
1740 PRINTCD$"                        ■"
1750 CH=VAL(CH$):LOOP WHILECH>63ORCH<0
1760 GOSUB2030:GOTO1260
1770 REM ****************************
1780 REM       PUT UP DISPLAY
1790 REM ****************************
1800 SCNCLR:PRINTCHR$(27)"M";:REM DISABLE SCROLL
1810 PRINT"■■==========          ■===■       CHARACTER"
1820 FORI=1TO8
1830 PRINT"■=■........■="
1840 NEXT
1850 PRINT"■=========="
1860 PRINT"■■■■■■■■■■■■■■■■■■■■■■= ="
1870 PRINT"■■■■■■■■■■■■■■■■■■■■■===="
1880 PRINT"■■■■■■■■■"SPC(20)"CONTROLS:"
1890 PRINTSPC(20)"+ ADD DOT    "
1900 PRINTSPC(20)"- DELETE DOT"
1910 PRINTSPC(20)"M MULTI ON   "
1920 PRINTSPC(20)"0 MULTI OFF "
1930 PRINTSPC(20)"S SELECT     "
1940 PRINTSPC(20)"N NEXT       "
1950 PRINTSPC(20)"Q QUIT       "
1960 PRINTSPC(22)"COLOURS:  "
1970 PRINTSPC(20)"F1 SCREEN    "
1980 PRINTSPC(20)"F2 BORDER    "
1990 PRINTSPC(20)"F3 CHARACTER"
2000 PRINTSPC(20)"1  M-C 1     "
2010 PRINTSPC(20)"2  M-C 2     ■"
2020 GOSUB2790
2030 PRINT"■■"SPC(29)"NUMBER"RIGHT$("     "+STR$(CH),4)
2040 FORR=0TO7::A=PEEK(14848+R+(CH)*8)
2050 FORC=0TO7
2060 IF(AANDP2(7-C))=0THENPOKEFNS(0),46:ELSEPOKEFNS(0),62
2070 NEXT:NEXT:POKE3132,CH+64:POKE2108,CC+CL*16
2080 R=0:C=0:RETURN
2090 REM ****************************
2100 REM LOAD A CHARACTER FILE FROM
2110 REM TAPE OR DISK
2120 REM ****************************
2130 PRINTCHR$(8)"□             CHARACTER EDITOR
2140 PRINT"■■DO YOU WISH TO LOAD A CHARACTER FILE ?";
2150 DO:DIRECTORY A$:LOOP DIRECTORY A$="Y" OR A$="N"
2160 IFA$="N"THEN RETURN
2170 PRINT"Y":GOSUB2470:REM SET UP FILE DETAILS
2180 POKE2034,0:SYS(65493)
2190 IF(PEEK(2037)AND1)=0THEN RETURN
2200 PRINT"■■LOAD ERROR- DO YOU WANT TO TRY AGAIN?"
2210 DO:DIRECTORYA$:LOOP DIRECTORY A$="Y"ORA$="N"
2220 IFA$="N"THEN RETURN
2230 GOTO2100
2240 REM ****************************
2250 REM END PROGRAM AND SAVE CHARS
2260 REM ****************************
```

```
2270 PRINT"■DO YOU WISH TO SAVE THE CHARACTERS ?";
2280 DO:DIRECTORY A$:LOOP DIRECTORYA$="Y"ORA$="N"
2290 IFA$="N"THEN2400
2300 PRINTA$
2310 SS=DEC("3A00"):ES=DEC("3C00")
2320 GOSUB2470
2330 POKE2034,3:POKE2036,INT(ES/256):POKE2035,ES-INT(ES/256)*256
2340 POKE4,INT(SS/256):POKE3,SS-INT(SS/256)*256
2350 SYS(65496):REM SAVE
2360 IFST=0THEN2400
2370 PRINT"■■SAVE ERROR- DO YOU WANT TO TRY AGAIN?"
2380 DO:DIRECTORY A$:LOOP DIRECTORYA$="Y"ORA$="N"
2390 IF A$="Y"THEN2250
2400 COLOR4,4,6:COLOR0,2,7
2410 PRINT"■■"
2420 POKETED+18,PEEK(TED+18)OR4:POKETED+19,(PEEK(TED+19)AND3)OR208
2430 POKETED+7,PEEK(TED+7)AND239
2440 PRINTCHR$(27)"L";:REM REENABLE SCROLL
2450 END
2460 REM *****************************
2470 REM GET FILE DETAILS FOR LOAD
2480 REM AND SAVE
2500 PRINT"■■PLEASE ENTER FILE NAME: ";
2510 OPEN1,0:INPUT#1,FM$:CLOSE1:PRINT
2520 PRINT"■■TAPE OR DISK: ";
2530 DO:DIRECTORYA$:DIRECTORYA$="T"ORA$="D"
2540 PRINTA$:D=8:IFA$="T"THEND=1
2550 IF LEN(FM$)>16THENFM$=LEFT$(FM$,16)
2560 POKE171,LEN(FM$)
2570 POKE172,D:POKE173,D:POKE174,D
2580 POKE175,0:POKE176,2
2590 FORI=1TOLEN(FM$):POKE511+I,ASC(MID$(FM$,I,1)):NEXT
2600 RETURN
2610 REM *****************************
2620 REM COPY CHARACTERS INTO RAM FOR
2630 REM EDITING
2640 REM *****************************
2650 A=14336:B=14848
2660 FORI=53248TO53759
2670 POKEA,PEEK(I):POKEB,PEEK(I):A=A+1:B=B+1
2680 FORI=0TO7
2690 POKE14336+61*8+I,255
2700 POKE14336+62*8+I,PEEK(53248+81*8+I)
2710 NEXT:RETURN
2720 REM *****************************
2730 REM CHANGE COLOUR OF USER DEF
2740 REM CHARACTERS
2750 REM *****************************
2760 POKE 2108,CC+CL*16
2770 GOSUB2880:GOTO1260
2780 REM *****************************
2790 REM PUT UP CHARACTER DISPLAY
2800 REM *****************************
2810 PRINT"■■■■■■■■■■■■■    0123456789 "
2820 PRINT"  ■============="
2830 FORI=0TO5
2840 PRINT"■"RIGHT$("    "+STR$(I),2);"■=            "
2850 NEXT
2860 PRINT"■ 6■=     ========"
2870 PRINT"  =======■■";
2880 FORRR=0TO5
2890 FORCR=0TO9
2900 POKE3595+RR*40+CR,RR*10+CR+64
2910 POKE2571+RR*40+CR,CC+CL*16
2920 NEXT:NEXT:FORI=0TO3
2930 POKE3835+I,I+124:POKE2811+I,CC+CL*16
2940 NEXT:RETURN
2950 REM *****************************
2960 REM ERROR TRAP ROUTINE
2970 REM *****************************
2980 IF ER=30 THEN RESUME:REM NO STOP KEY
2990 PRINT:PRINT"■?"ERR$(ER)" ERROR IN"EL;
3000 COLOR0,2,7:COLOR4,4,6:GOTO2420
3010 REM
3020 REM
3030 DATA 133,134,135,137,138,139
```

Testing Ground

Eric Corbett brings you his Basic Test System, to make writing in Basic a little less of a headache.

THERE ARE VERY FEW EFFECTIVE program development aids available for use on home micros. This program provides, what I think, is one of the most useful – an interactive testing system.

This Basic Interactive Test System (BTS) creates a test harness which allows the efficient debugging of Basic programs running under its control. Functions available include single-stepping, setting of break points and line tracing.

It's a machine code routine which should prove invaluable when writing Basic programs on your C64.

Using the System

Load and run BTS before beginning Basic programming. After a short delay, during which BTS will be locating itself from 49152 to 50432, the message

READY

will appear on the screen. Type in NEW and the machine may now be used as normal.

The program listing as provided incorporates a check sum at the end of each 128 bytes of data. If a typing error has been made, then a check sum discrepancy will be reported for the block of data in question. This must be corrected before BTS can be used.

Commands Available

To enter into a dialogue with BTS, press function key 1. The message

BTS READY
>

will appear on the screen. The prompt character (>) indicates that the machine is now in the BTS command mode. To leave this mode and return to the Basic immediate mode, type END.

In the command mode, the following commands are available: FAST, SLOW, TRACE, STEP, BREAK

BREAK

This allows a break-point to be set at any line in the program. This is equivalent to inserting a Basic STOP verb at the start of the break-pointed line, but is much more powerful because the source program is not actually altered.

For example:

>BREAK 150

would cause program execution to be halted when line 150 was about to be obeyed.

Having thus halted the program at the desired line, values of variables may be checked or altered as usual at a break-point.

The program may be continued normally from this point (by use of the Basic CONT verb). However, two extra functions are available through BTS. Pressing function key two will cause the break-pointed line to be listed to the screen. Pressing function key seven will continue execution of the Basic program.

Typing in

>BREAK 0

will clear the break-point.

STEP

This command causes the Basic program to be executed in a single-step mode, i.e. a break-point is set for every line in the program.

Thus at each new line about to be obeyed, BTS will force a return to the Basic immediate mode. From here execution may be continued as described under the BREAK command. This allows very close monitoring of the flow through a program.

For example:

>STEP ON

would enable the single-step mode.

>STEP OFF

would disable the single-step mode.

TRACE

When the TRACE mode is set, BTS displays on the screen the line number of each line in the program as it is obeyed in the form [150], for example. This is equivalent to the TRON/TROFF commands on the C128.

For example:

>TRACE ON

would enable the line trace mode.

>TRACE OFF

would disable the line trace mode.

SLOW

This command slows down the speed of execution of Basic programs by a factor of approximately five. This can be particularly useful – for instance when using the line tracing mode – as otherwise it is very difficult to follow what is happening when the line numbers are changing at normal speed.

FAST

This command returns the execution speed of Basic programs to normal.

PROGRAM: BTS

```
10 REM ====================
==========
20 REM  BASIC INTERACTIVE TE
ST SYSTEM
30 REM
40 REM       ERIC CORBETT 19
85
50 REM ====================
==========
60 GOSUB 240
70 P=49152:HA=0:ER=0
80 FOR J=1 TO 10
90 FOR K=0 TO 127
100 READ A
110 POKE P+K,A
120 HA=HA+A
130 NEXT K
140 HA=HA-(INT(HA/256)*256)
150 READ A
160 IF A=HA THEN 190
170 PRINT "ERROR IN DATA BLO
CK";J
180 ER=1
190 HA=0
200 P=P+128
210 NEXT J
220 IF ER=0 THEN SYS 49285
230 END:GOTO 230
240 REM ====================
==========
250 REM   CLEAR SCREEN AND
DISPLAY
260 REM        INITIAL MESSAG
E.
270 REM ====================
==========
280 PRINT CHR$(147)
290 PRINT CHR$(18);
300 PRINT TAB(4)" BASIC INTE
RACTIVE ";
310 PRINT "TEST SYSTEM "
320 FOR I=1 TO 3
330 PRINT CHR$(17)
340 NEXT
350 PRINT TAB(14)"INITIALISI
NG"
```

```
360 PRINT TAB(14)"-PLEASE WAIT"
370 PRINT
380 RETURN
390 REM *** DATA BLOCK 1 ***
400 DATA 0,0,0,141,2,192,142,1
410 DATA 192,140,0,192,169,13,72,160
420 DATA 16,169,0,6,251,38,252,42
430 DATA 201,10,144,4,233,10,230,251
440 DATA 136,208,240,201,10,144,2,105
450 DATA 6,105,48,72,165,251,5,252
460 DATA 208,221,160,0,104,201,13,240
470 DATA 6,200,145,253,76,52,192,169
480 DATA 93,200,145,253,169,0,200,145
490 DATA 253,172,0,192,174,1,192,173
500 DATA 2,192,96,120,165,251,141,20
510 DATA 3,165,252,141,21,3,88,96
520 DATA 105,0,0,0,0,0,25,79
530 DATA 0,0,0,91,0,0,0,0
540 DATA 0,0,0,230,122,208,2,230
550 DATA 123,96,198,122,165,122,201,255
560 DATA 131
570 REM      *** DATA BLOCK 2 ***
580 DATA 208,2,198,123,96,162,2,189
590 DATA 158,192,149,115,202,16,248,32
600 DATA 202,193,169,0,141,106,192,141
610 DATA 99,192,141,100,192,96,76,161
620 DATA 192,32,115,192,142,96,192,140
630 DATA 97,192,141,98,192,165,123,201
640 DATA 8,144,3,76,185,192,76,158
650 DATA 193,173,105,192,240,17,173,106
660 DATA 192,240,7,173,104,192,201,1
670 DATA 208,5,169,0,141,104,192,173
680 DATA 104,192,208,32,165,57,205,102
690 DATA 192,208,10,165,58,205,103,192
700 DATA 208,3,76,158,193,173,101,192
710 DATA 240,10,162,127,160,255,136,208
720 DATA 253,202,208,248,165,57,141,102
730 DATA 192,165,58,141,103,192,173,105
740 DATA 187
750 REM      *** DATA BLOCK 3 ***
760 DATA 192,240,10,165,57,141,99,192
770 DATA 165,58,141,100,192,173,99,192
780 DATA 205,102,192,208,77,173,100,192
790 DATA 205,103,192,208,69,173,104,192
800 DATA 208,69,32,122,192,32,122,192
810 DATA 32,122,192,32,122,192,32,122
820 DATA 192,165,122,133,251,165,123,133
830 DATA 252,160,0,177,251,240,18,32
840 DATA 115,192,32,115,192,32,115,192
850 DATA 32,115,192,32,115,192,76,103
860 DATA 193,165,122,133,61,165,123,133
870 DATA 62,169,255,141,104,192,56,76
880 DATA 65,168,169,0,141,104,192,173
890 DATA 106,192,240,50,173,104,192,201
900 DATA 1,240,43,173,102,192,133,251
910 DATA 173,103,192,133,252,169,107,133
920 DATA 191
930 REM      *** DATA BLOCK 4 ***
940 DATA 253,169,192,133,254,32,3,192
950 DATA 162,0,189,107,192,32,210,255
960 DATA 232,189,107,192,208,247,173,104
970 DATA 192,41,1,141,104,192,174,96
980 DATA 192,172,97,192,173,98,192,76
990 DATA 121,0,0,0,0,67,79,78
1000 DATA 84,13,0,76,73,83,84,0
1010 DATA 0,0,0,0,0,66,84,83
1020 DATA 32,82,69,65,68,89,46,13
1030 DATA 0,0,141,172,193,169,220,133
1040 DATA 251,169,193,133,252,32,83,192
1050 DATA 173,172,193,96,142,170,193,140
1060 DATA 171,193,141,172,193,32,159,255
1070 DATA 166,198,208,3,76,111,194,202
1080 DATA 189,119,2,201,133,208,36,134
1090 DATA 198,160,255,200,185,189,193,240
1100 DATA 15
1110 REM      *** DATA BLOCK 5 ***
1120 DATA 6,32,210,255,76,251,193,32
1130 DATA 163,194,160,3,200,185,189,193
1140 DATA 240,6,32,210,255,76,12,194
1150 DATA 76,111,194,201,136,208,17,160
1160 DATA 255,200,232,185,173,193,157,119
1170 DATA 2,208,246,134,198,76,111,194
1180 DATA 201,135,208,59,173,102,192,133
1190 DATA 251,173,103,192,133,252,169,179
1200 DATA 133,253,169,193,133,254,24,165
1210 DATA 253,105,3,133,253,165,254,105
1220 DATA 0,133,254,32,3,192,160,255
1230 DATA 200,232,185,179,193,157,119,2
1240 DATA 201,93,208,244,169,13,157,119
1250 DATA 2,232,134,198,76,111,194,174
1260 DATA 170,193,172,171,193,173,172,193
1270 DATA 76,49,234,0,0,0,0,0
1280 DATA 132
1290 REM      *** DATA BLOCK 6 ***
1300 DATA 3,2,5,66,82,69,65,75
1310 DATA 3,69,78,68,4,70,65,83
1320 DATA 84,4,83,76,79,87,4,83
1330 DATA 84,69,80,5,84,82,65,67
1340 DATA 69,62,63,173,161,194,32,210
1350 DATA 255,160,255,200,32,207,255,201
1360 DATA 17,240,248,201,145,240,244,153
1370 DATA 0,2,201,13,240,3,76,171
1380 DATA 194,140,123,194,32,210,255,169
1390 DATA 1,141,125,194,160,130,132,251
1400 DATA 160,194,132,252,32,12,195,144
1410 DATA 3,76,100,195,173,125,194,201
1420 DATA 7,144,14,173,162,194,32,210
1430 DATA 255,169,13,32,210,255,76,163
1440 DATA 194,238,125,194,24,160,0,177
1450 DATA 251,101,251,133,251,169,0,101
1460 DATA 67
1470 REM      *** DATA BLOCK 7 ***
1480 DATA 252,133,252,230,251,208,205,230
1490 DATA 252,76,212,194,169,0,133,253
1500 DATA 169,2,133,254,160,0,177,251
1510 DATA 141,124,194,206,124,194,165,251
1520 DATA 72,165,252,72,230,251,208,2
1530 DATA 230,252,169,0,141,128,194,141
1540 DATA 129,194,172,128,194,177,253,201
1550 DATA 32,208,7,200,140,128,194,76
1560 DATA 50,195,200,140,128,194,172,129
1570 DATA 194,209,251,208,15,204,124,194
1580 DATA 144,3,76,93,195,200,140,129
1590 DATA 194,76,50,195,24,104,133,252
1600 DATA 104,133,251,96,238,124,194,173
1610 DATA 125,194,201,1,240,19,201,2
```

```
1620 DATA 240,36,201,3,240,4
4,201,4
1630 DATA 240,59,201,5,240,7
4,76,254
1640 DATA 108
1650 REM      *** DATA BLOCK
8 ***
1660 DATA 195,32,95,196,208,
3,76,227
1670 DATA 194,165,251,141,99
,192,165,252
1680 DATA 141,100,192,76,163
,194,173,124
1690 DATA 194,205,123,194,24
0,3,76,220
1700 DATA 194,96,173,124,194
,205,123,194
1710 DATA 240,3,76,220,194,1
69,0,141
1720 DATA 101,192,76,163,194
,173,124,194
1730 DATA 205,123,194,240,3,
76,220,194
1740 DATA 169,255,141,101,19
2,76,163,194
1750 DATA 32,28,196,201,0,20
8,3,76
1760 DATA 227,194,201,1,208,
20,169,255
1770 DATA 141,105,192,173,99
,192,141,126
1780 DATA 194,173,100,192,14
1,127,194,76
1790 DATA 163,194,169,0,141,
105,192,173
1800 DATA 126,194,141,99,192
,173,127,194
1810 DATA 141,100,192,76,163
,194,32,28
1820 DATA 24
1830 REM      *** DATA BLOCK
9 ***
1840 DATA 196,201,0,208,3,76
,227,194
1850 DATA 201,1,208,8,169,25
5,141,106
1860 DATA 192,76,163,194,169
,0,141,106
1870 DATA 192,76,163,194,172
,128,194,177
1880 DATA 253,201,32,208,4,2
00,76,31
1890 DATA 196,201,79,208,45,
200,177,253
1900 DATA 201,78,208,10,200,
177,253,201
1910 DATA 13,240,26,76,90,19
6,201,70
1920 DATA 208,24,200,177,253
,201,70,208
1930 DATA 17,200,177,253,201
,13,208,10
1940 DATA 169,2,76,92,196,16
9,1,76
1950 DATA 92,196,169,0,96,0,
0,169
1960 DATA 0,133,251,133,252,
172,128,194
1970 DATA 177,253,201,32,208
,4,200,76
1980 DATA 104,196,201,13,240
,73,201,48
1990 DATA 144,69,201,58,176,
65,56,233
2000 DATA 142
2010 REM      *** DATA BLOCK
10 ***
2020 DATA 48,72,6,251,38,252
,165,251
2030 DATA 141,93,196,165,252
,141,94,196
2040 DATA 6,251,38,252,6,251
,38,252
2050 DATA 24,165,251,109,93,
196,133,251
2060 DATA 165,252,109,94,196
,133,252,104
2070 DATA 24,101,251,133,251
,165,252,105
2080 DATA 0,133,252,200,177,
253,201,13
2090 DATA 208,188,169,255,76
,193,196,169
2100 DATA 0,96,193,3,196,3,2
13,3
2110 DATA 222,3,229,3,238,3,
5,4
2120 DATA 13,4,16,4,26,4,38,
4
2130 DATA 45,4,59,4,62,4,73,
4
2140 DATA 74,4,74,4,74,4,74,
4
2150 DATA 80,4,80,4,80,4,80,
4
2160 DATA 89,4,135,4,150,4,1
65,4
2170 DATA 175,4,200,4,224,4,
225,4
2180 DATA 200
```



TOPMON

D Chiles provides an extremely fancy machine code monitor.

ONE OF THE MOST USEFUL programming tools that is available to the machine code programmer is a monitor and many are available on the market. Topmon, however, is probably one of the most powerful monitors you will come across, offering the user over 20 commands including disassembly and the ability to enter text into memory.

Exactly why is a machine code monitor so useful? As you probably already know, machine code programs are stored in memory as a series of numbers. These numbers can be seen from Basic by using the PEEK instruction. If you want to see an example of this then enter the following program:

```
10 FOR C=40960 TO 40970
20 PRINT PEEK (C)
30 NEXT C
```

All this program does is to print out the first 10 numbers of the Basic ROM. In case you didn't know, Basic is a machine code program that runs automatically when you turn your computer on. When you enter any commands it is this machine code program that works out exactly what it is you want to do and then performs the action. This program is stored in memory from location 40960 to 49151.

Obviously PEEKing a series of locations every time you want to look at them would be a fairly long winded process. Altering a series of memory locations with the POKE function would take even more time. This is where the machine code monitor comes to the rescue.

With topmon the command 'M' allows you to display and change (Modify) a series of locations. The format of the command is:

M ssss eeee

where ssss and eeee are the start and end locations which you wish to look at. However one small point must be remembered, most machine code monitors, including Topmon, require that any numbers be entered in hexadecimal format (i.e. Base 16). Using hexadecimal makes life very easy for the machine code programmer as the later example will show.

```
PROGRAM:  TOPMON LOAD

1000 REM ********************************
1010 REM *[SPC4]TOPMON BASIC LOADER[SPC5]*
1020 REM ********************************
2000 FOR L=0 TO 258:CX=0:FOR D=0 TO 15:READ A:CX=CX+A
   :POKE 2049+L*16+D,A:NEXT D
2010 READ A:IF A<>CX THEN PRINT"ERROR IN LINE";2040+(L*10)
   :STOP
2020 NEXT L:END
2040 DATA 69,8,0,0,158,50,51,57,53,58,143,34,20,20,20,20,
   761
2050 DATA 20,20,20,20,20,20,20,20,20,20,42,42,32,32,85,32,
   465
2060 DATA 32,76,32,32,84,32,32,82,32,32,65,32,32,77,32,32,
   736
2070 DATA 79,32,32,78,32,32,45,32,32,54,32,32,52,32,32,42,
   670
2080 DATA 42,160,160,0,0,0,1,0,153,34,147,144,85,76,84,82,
   1168
2090 DATA 65,77,79,78,45,54,52,46,46,46,66,89,32,68,46,67,
   956
2100 DATA 72,73,76,69,83,34,58,153,34,183,183,183,183,183,
   183,183,1933
2110 DATA 183,183,183,183,183,183,183,183,183,183,183,183,
   183,183,183,183,2928
2120 DATA 183,183,34,0,179,8,2,0,153,34,5,213,192,201,17,
   157,1561
2130 DATA 157,157,221,144,67,5,221,17,157,157,157,202,192,
   203,145,68,2270
2140 DATA 46,67,72,73,76,69,83,32,32,32,49,57,56,53,17,17,
   831
2150 DATA 34,0,221,8,3,0,153,34,17,17,144,67,79,77,77,65,
   996
2160 DATA 78,68,83,32,58,78,77,82,71,70,80,76,83,35,36,58,
   1065
2170 DATA 62,84,64,88,66,86,72,42,68,95,34,0,37,9,4,0,811
2180 DATA 133,34,19,17,17,17,17,17,17,80,76,69,65,83,69,32,
```

Any memory location can only hold a number up to 255. Therefore, if we want to store a number that is greater than 255 we need to use two memory locations and store the high part of the number in one location and the low byte in the second. If we wanted to break down a decimal number into its two parts we would have to use the following formula:

Low number =number-INT(number/256)*256
High number=INT(number/256)

To make things even more awkward the number is stored with the low part in the lower memory location and the high part in the following location. For example, the number 49152 will break down into 192 and zero. Use the above formula if you want to work this out for yourself.

If you then want to find out what number two addresses contain in decimal you would have to use the following formula. Complicated isn't it?

number=PEEK(low)+256*PEEK (high)

Now let's take a look at hexadecimal. If we wanted to break a hexadecimal number down in to two parts there are no complicated formulae. All we need to do is split the number in half — e.g. the hexadecimal number F36A breaks down into:

```
   762
2190 DATA 69,78,84,69,82,32,80,65,71,69,32,78,79,46,32,40,
   1006
2200 DATA 57,45,49,57,53,41,32,34,59,80,58,139,80,179,57,
   176,1196
2210 DATA 80,177,49,57,53,176,80,177,49,52,55,175,80,179,
   49,57,1545
2220 DATA 50,167,49,0,89,9,5,0,153,34,147,5,82,69,45,69,973
2230 DATA 78,84,82,89,32,80,79,73,78,84,32,61,34,40,80,170,
   1176
2240 DATA 49,41,172,50,53,54,34,144,34,58,151,50,52,48,54,
   44,1088
2250 DATA 80,58,158,50,52,48,55,0,0,0,169,133,141,69,8,169,
   1190
2260 DATA 8,141,70,8,96,148,169,13,133,80,169,0,133,76,169,
   22,1435
2270 DATA 133,77,174,102,9,160,0,177,76,133,78,200,177,76,
   240,19,1831
2280 DATA 133,79,200,138,145,78,165,76,24,105,2,133,76,144,
   230,230,1958
2290 DATA 77,208,226,232,198,80,208,238,169,160,205,102,9,
   176,3,141,2432
2300 DATA 230,9,160,0,169,255,133,76,133,78,169,21,133,77,
   173,102,1918
2310 DATA 9,170,24,105,12,133,79,177,76,145,78,165,76,56,
   233,1,1539
2320 DATA 133,76,133,78,176,241,198,79,198,77,165,77,201,8,
   208,231,2279
2330 DATA 232,142,219,9,32,229,9,234,76,0,149,162,0,44,162,
   1,1700
2340 DATA 142,134,2,96,169,148,133,56,169,0,133,55,76,68,
   166,32,1579
2350 DATA 220,148,76,75,0,32,220,148,32,29,151,76,223,148,
   0,8,1586
2360 DATA 216,72,138,72,152,72,169,37,141,22,3,169,193,141,
   23,3,1623
2370 DATA 32,20,199,234,234,234,234,234,234,169,128,133,
   157,104,168,104,2618
2380 DATA 170,104,40,0,162,0,104,157,1,192,232,224,4,208,
   247,104,1949
2390 DATA 56,233,1,141,5,192,104,233,0,141,6,192,186,142,0,
   192,1824
2400 DATA 234,234,234,234,234,234,234,234,234,234,234,32,
   220,192,234,216,3468
2410 DATA 162,34,189,66,194,32,171,203,202,16,247,173,6,
   192,172,5,2064
2420 DATA 192,32,48,194,173,4,192,41,4,240,3,32,104,196,
   173,21,1649
2430 DATA 3,172,20,3,32,128,194,32,24,195,162,4,189,0,192,
   32,1382
2440 DATA 249,194,32,24,195,202,16,244,169,32,168,32,29,
   195,162,7,1950
2450 DATA 169,128,72,160,45,45,4,192,240,3,188,58,194,152,
   32,171,1853
2460 DATA 203,104,74,202,16,236,32,20,195,162,255,154,169,
   13,160,46,2041
2470 DATA 32,246,192,32,207,255,201,46,240,249,201,32,240,
   245,201,13,2632
2480 DATA 240,234,162,24,221,231,193,240,6,202,16,248,76,
   104,194,138,2529
2490 DATA 10,170,189,0,194,133,76,189,1,194,133,77,169,76,
   133,75,1819
2500 DATA 32,240,192,24,144,198,78,77,82,71,70,80,76,83,36,
   35,1518
2510 DATA 58,95,84,64,88,66,86,72,42,63,68,62,0,0,0,20,868
2520 DATA 199,113,196,74,199,48,197,112,197,134,198,39,196,
   0,196,174,2272
2530 DATA 198,192,198,236,196,184,200,154,198,108,198,247,
   203,236,197,43,2988
2540 DATA 196,87,199,236,200,218,200,40,201,0,204,104,194,
   104,194,32,2409
2550 DATA 249,194,152,32,249,194,76,24,195,67,90,73,68,66,
   45,86,1860
2560 DATA 78,32,32,95,46,13,80,83,32,82,89,32,82,88,32,67,
   963
2570 DATA 65,32,82,83,32,32,81,82,73,32,32,67,80,32,32,32,
   869
2580 DATA 32,160,13,13,162,0,44,162,17,44,162,24,44,162,33,
   44,1116
2590 DATA 162,42,134,75,32,220,192,234,234,162,250,154,76,
   215,197,32,2411
2600 DATA 249,194,152,32,249,194,76,108,196,169,128,133,
   157,162,1,134,2334
2610 DATA 185,162,0,96,83,84,65,82,84,47,69,78,68,32,65,68,
   1268
2620 DATA 68,82,69,83,211,67,79,77,77,65,78,196,84,69,88,
   84,1477
2630 DATA 47,78,65,77,197,67,72,65,82,65,67,84,69,210,84,
   65,1394
2640 DATA 80,69,47,68,73,83,75,32,69,82,82,79,210,32,222,
   194,1497
2650 DATA 10,10,10,10,133,75,32,222,194,24,101,75,96,32,21,
   196,1241
2660 DATA 56,233,48,144,12,201,10,144,14,201,17,144,4,201,
   23,144,1596
2670 DATA 3,76,110,194,56,233,7,96,72,74,74,74,74,32,7,195,
   1377
2680 DATA 104,41,15,76,7,195,201,10,144,3,24,105,7,24,105,
   48,1109
2690 DATA 76,171,203,169,13,208,2,169,32,76,171,203,32,171,
   203,152,2051
2700 DATA 76,171,203,234,234,234,234,234,234,234,234,234,
   234,234,32,20,3076
2710 DATA 195,169,32,160,39,145,209,136,16,251,96,32,92,
   195,32,206,2005
2720 DATA 194,133,79,32,206,194,133,78,165,79,197,77,144,
   10,208,234,2163
2730 DATA 165,78,197,76,240,228,176,226,76,101,194,32,206,
   194,133,77,2399
2740 DATA 32,206,194,133,76,96,165,78,56,229,76,133,78,165,
   79,229,2025
2750 DATA 77,133,79,162,3,70,79,102,78,202,208,249,230,78,
   230,79,2059
2760 DATA 96,32,140,195,201,13,208,26,76,107,194,162,0,134,
   81,164,1829
2770 DATA 211,177,209,201,59,240,77,201,32,208,13,32,207,
   255,201,13,2336
2780 DATA 208,237,134,80,202,48,225,96,201,34,208,24,32,
```

```
   207,255,32,2223
2790 DATA 207,255,201,34,240,217,201,13,240,206,157,9,192,
   232,224,75,2703
2800 DATA 208,237,240,196,201,63,240,13,32,206,194,157,9,
   192,232,224,2644
2810 DATA 75,208,188,240,179,164,81,138,153,91,192,200,132,
   81,32,207,2361
2820 DATA 255,24,144,231,134,80,76,207,255,32,130,195,32,
   206,194,133,2328
2830 DATA 186,165,80,162,9,160,192,32,189,255,169,1,133,
   185,96,32,2046
2840 DATA 234,195,32,60,195,32,20,195,166,78,164,79,169,76,
   32,216,1943
2850 DATA 255,76,69,196,32,207,255,201,44,240,249,201,32,
   240,245,201,2743
2860 DATA 13,208,219,76,110,194,169,0,240,2,169,1,72,32,
   138,194,1837
2870 DATA 134,183,232,134,186,234,32,207,255,201,13,240,3,
   32,234,195,2515
2880 DATA 104,32,213,255,176,19,165,144,162,0,134,144,166,
   186,202,208,2310
2890 DATA 4,41,48,208,4,41,191,240,165,76,113,194,32,20,
   195,169,1741
2900 DATA 63,160,32,76,29,195,0,169,18,208,2,169,146,76,
   171,203,1717
2910 DATA 32,60,195,32,103,195,32,36,195,169,46,160,58,32,
   29,195,1569
2920 DATA 32,32,197,160,0,162,8,32,24,195,177,76,32,249,
   194,200,1770
2930 DATA 202,208,244,169,34,32,171,203,160,0,162,8,177,76,
   32,191,2069
2940 DATA 196,200,202,208,247,165,76,24,105,8,144,2,230,77,
   133,76,2093
2950 DATA 198,78,208,4,198,79,240,74,32,4,197,24,144,184,
   201,34,1899
2960 DATA 240,36,201,5,240,34,201,129,240,30,201,17,144,24,
   201,20,1963
2970 DATA 144,22,201,28,144,16,201,128,144,14,201,133,144,
   8,201,141,1870
2980 DATA 144,6,201,144,176,2,169,46,76,171,203,32,92,195,
   162,8,1827
2990 DATA 160,0,32,206,194,145,76,32,207,255,201,13,240,4,
   200,202,2167
3000 DATA 208,240,96,32,228,255,201,3,240,15,201,32,208,
   244,32,228,2463
3010 DATA 255.201,32,240,237,201,3,208,245,234,234,234,76.
   73,204,165,2842
3020 DATA 77,32,249,194,165,76,76,249,194,234,234,234,234,
   234,234,32,2748
3030 DATA 207,255,133,78,201,13,240,3,32,92,195,174,0,192,
   154,173,2142
3040 DATA 4,192,72,173,3,192,174,2,192,172,1,192,72,169,13,
   197,1820
3050 DATA 78,240,5,104,40,108,76,0,104,40,108,5,192,166,88,
   202,1556
3060 DATA 208,5,32,36,195,162,7,134,88,32,32,197,76,24,195,
   32,1455
3070 DATA 60,195,32,140,195,160,1,132,88,32,36,195,166,80,
   160,0,1672
```

High Part F3 and
Low part 6A

This is much easier. Converting back to the larger number is also easier since all you have to do is read them back from memory and put the two numbers in the correct order.

Now for a practical example. Memory locations 43 and 44 (2B and 2C hex) hold the start address of your Basic program. Now, if we use the 'M' command of Topmon, we can see where your Basic program starts. All you need to enter is —

M 002B 002C

Topmon will then display a line of numbers and digits. This line will consist of a series of eight hexadecimal numbers and, at the end, the ASCII representation of the numbers (i.e. what character that number represents. If no character then a '.' is displayed). The first two numbers of this line are the numbers that are stored in locations 2B and 2C. These will be 01 and 08. Putting the numbers in the correct order, we can tell that the Basic program starts at 0B01 hex. Simple isn't it?

If you want to have a look at the section of memory that we PEEKed at the beginning of this article then you will need to enter the command —

M A000 A000A

The 'M' command not only allows you to display memory but will also allow you to change it. If you move the cursor to any number which you wish to change and enter the new number on top of the old (in hexadecimal) and then press RETURN, the new number will be stored in the relevant memory location. Don't forget that the change will not take place until you press RETURN.

Now that we've got the basics out of the way let's have a look at the other commands that are available from Topmon.

Enter Values — :

The : command allows you to enter up to eight hexadecimal values into memory. The command has the following format:

:ssss xx xx xx..up to 8 xx's

where ssss is the address where you want to store the numbers and xx are the numbers you want to save in memory. The "M" instruction places a ':' at the start of each line and this is why you are able to edit each line so easily .

Normalise — N

When developing programs in machine code you will quite often end up with weird screen colours and perhaps even notes that carry on playing. Entering 'N' when you are in the monitor will normalise the screen colours and switch off any sounds that are playing.

Registers — R

This command will display the contents of the microprocessor's internal registers. All the microprocessor's flags are also shown as letters. This instruction is very useful for examining what state the microprocessor is in at any time. The R command places a ← before the registers so that they can be changed by the following command.

← Enter Registers

This command allows you to alter any of the microprocessoror's internal registers and takes the following format:

← pppp iiii ss aa xx yy ss

The registers fall as indicated below:

pc irq sp ar xr yr sr
pppp iiii ss aa xx yy ss

Relocate — R

This command has the same letter as the Register but has a number of parameters with it. The format for this command is:

R aaaa bbbb cccc dddd eeee

Relocate is quite a complicated and very powerful command. It allows you to make any program in memory run somewhere else.

```
3080 DATA 132,82,165,81,197,82,240,17,144,15,152,164,82,
   217,91,192,2053
3090 DATA 208,6,168,230,82,208,9,234,168,177,76,217,9,192,
   208,7,2199
3100 DATA 200,202,208,222,32,94,197,32,4,197,166,80,32,179,
   197,24,2066
3110 DATA 144,204,230,76,208,4,230,77,240,5,32,195,197,144,
   20,76,2082
3120 DATA 170,193,165,79,197,77,144,12,208,8,165,78,197,76,
   240,2,2011
3130 DATA 144,2,24,96,56,96,32,93,196,166,75,189,149,194,
   72,41,1625
3140 DATA 127,32,171,203,232,104,16,243,76,170,193,32,60,
   195,32,206,2092
3150 DATA 194,133,81,32,206,194,133,80,165,77,197,81,144,
   29,208,6,1960
3160 DATA 165,76,197,80,144,21,160,0,177,76,145,80,32,179,
   197,230,1959
3170 DATA 80,208,2,230,81,32,195,197,144,238,96,165,78,56,
   229,76,2107
3180 DATA 133,82,165,79,229,77,133,83,165,80,24,101,82,133,
   80,165,1811
3190 DATA 81,101,83,133,81,160,0,177,78,145,80,165,78,56,
   233,1,1652
3200 DATA 133,78,176,8,198,79,165,79,201,255,240,156,165,
   80,56,233,2302
3210 DATA 1,133,80,176,8,198,81,165,81,201,255,240,139,234,
   234,234,2460
3220 DATA 234,234,234,234,234,32,195,197,144,205,96,32,92,
   195,173,136,2667
3230 DATA 2,133,79,160,0,132,78,177,76,145,78,173,134,2,
   153,0,1522
3240 DATA 216,200,208,243,96,32,60,195,32,206,194,133,80,
   160,0,165,2220
3250 DATA 80,145,76,32,179,197,144,247,96,32,92,195,32,140,
   195,232,2114
3260 DATA 234,160,0,185,9,192,145,76,200,202,208,247,96,32,
   92,195,2273
3270 DATA 169,32,160,61,32,29,195,166,76,165,77,76,219,203,
   234,165,2059
3280 DATA 209,24,101,211,133,122,165,210,105,0,133,123,162,
   255,232,32,2217
3290 DATA 207,255,201,13,240,15,201,32,240,245,201,48,144,
   4,201,58,2305
3300 DATA 144,236,76,110,194,224,0,240,249,224,6,176,245,
   32,228,203,2587
3310 DATA 234,234,234,169,61,32,171,203,165,82,133,122,165,
   83,133,123,2344
3320 DATA 165,21,32,249,194,165,20,32,249,194,160,5,162,2,
   132,122,1904
3330 DATA 134,123,96,216,169,27,141,17,208,169,0,141,21,
   208,162,24,1856
3340 DATA 157,0,212,202,16,250,169,200,141,22,208,169,21,
   141,24,208,2140
3350 DATA 169,9,141,32,208,169,11,141,33,208,169,151,141,0,
   221,169,1972
3360 DATA 4,141,136,2,76,220,192,234,234,32,207,255,201,13,
   208,3,2158
3370 DATA 76,80,193,56,176,1,24,234,234,234,8,32,60,195,32,
   4,1639
3380 DATA 200,32,60,195,32,4,200,162,7,104,149,76,202,16,
   250,40,1729
3390 DATA 8,144,10,32,206,194,133,90,32,206,194,133,89,32,
   20,195,1718
3400 DATA 169,1,133,88,160,0,177,76,217,26,200,240,9,200,
   192,151,2039
3410 DATA 208,246,162,1,208,94,162,1,192,29,144,88,232,192,
   103,144,2206
3420 DATA 83,232,160,2,177,76,197,81,144,72,208,12,136,177,
   76,197,2030
3430 DATA 80,144,65,240,18,200,177,76,197,83,144,11,208,52,
   136,177,2008
3440 DATA 76,197,82,240,2,176,43,40,8,144,36,160,1,177,76,
   56,1514
3450 DATA 229,80,133,86,200,177,76,229,81,133,87,165,89,24,
   101,86,1976
3460 DATA 136,145,76,165,90,101,87,200,145,76,234,234,234,
   234,234,32,2423
3470 DATA 94,197,162,3,32,179,197,202,208,250,32,4,197,24,
   144,132,2057
3480 DATA 0,0,0,104,168,104,170,165,76,72,165,77,72,165,78,
   72,1488
3490 DATA 165,79,72,138,72,152,72,96,0,0,24,216,88,184,202,
   136,1696
3500 DATA 232,200,234,72,8,104,40,64,96,56,248,120,170,168,
   186,138,2136
3510 DATA 154,152,10,74,42,106,101,37,6,36,197,228,196,198,
   69,230,1836
3520 DATA 165,166,164,70,5,38,102,229,133,134,132,105,41,
   201,224,192,2101
3530 DATA 73,169,162,160,9,233,97,33,193,65,161,1,225,129,
   113,49,1872
3540 DATA 209,81,177,17,241,145,117,53,22,213,214,85,246,
   181,180,86,2267
3550 DATA 21,54,118,245,149,148,144,176,240,48,208,16,80,
   112,182,150,2091
3560 DATA 109,45,14,44,205,236,204,206,77,238,76,32,173,
   174,172,78,2083
3570 DATA 13,46,110,237,141,142,140,125,61,30,221,222,93,
   254,189,188,2212
3580 DATA 94,29,62,126,253,157,121,57,217,89,185,190,25,
   249,153,108,2115
3590 DATA 0,0,0,0,0,0,0,32,92,195,141,5,192,165,77,141,1040
3600 DATA 6,192,32,92,195,141,20,3,165,77,141,21,3,162,4,
   32,1286
3610 DATA 206,194,157,0,192,202,16,247,96,32,140,195,162,0,
   189,9,2037
3620 DATA 192,32,171,203,232,228,80,208,245,96,0,169,1,77,
   235,200,2369
3630 DATA 141,235,200,240,14,32,33,201,32,186,255,32,192,
   255,162,1,2211
3640 DATA 76,201,255,32,20,195,169,1,32,195,255,76,204,255,
   234,32,2232
3650 DATA 130,195,162,0,189,9,192,32,171,203,232,228,80,
   208,245,96,2372
3660 DATA 162,4,160,0,132,183,96,32,60,195,32,36,195,169,
   46,160,1662
3670 DATA 59,32,29,195,234,32,32,197,32,24,195,162,1,160,0,
```

```
   177,1561
3680 DATA 76,217,26,200,240,7,200,192,151,208,246,240,10,
   192,29,144,2378
3690 DATA 6,232,192,103,144,1,232,134,80,132,82,185,16,202,
   133,81,1955
3700 DATA 160,0,177,76,32,249,194,32,24,195,200,196,80,208,
   243,169,2235
3710 DATA 20,56,229,211,170,169,32,32,171,203,202,208,250,
   166,81,189,2389
3720 DATA 0,203,32,171,203,189,57,203,32,171,203,189,114,
   203,32,171,2173
3730 DATA 203,32,24,195,165,82,160,11,190,225,202,217,240,
   202,144,5,2297
3740 DATA 136,16,245,162,0,189,168,202,168,201,3,240,4,176,
   74,144,2128
3750 DATA 64,165,76,133,86,165,77,133,87,160,1,177,76,201,
   128,176,1905
3760 DATA 19,165,86,24,105,2,144,2,230,87,24,113,76,144,2,
   230,1453
3770 DATA 87,24,144,20,73,255,133,85,165,86,24,105,1,144,2,
   230,1578
3780 DATA 87,56,229,85,176,2,198,87,168,165,87,32,48,194,
   232,208,2054
3790 DATA 180,177,76,32,249,194,232,208,172,32,171,203,201,
   32,208,246,2613
3800 DATA 166,80,32,179,197,202,208,250,32,4,197,76,43,201,
   0,10,1877
3810 DATA 13,14,15,16,21,22,25,26,33,35,36,37,38,41,42,44,
   458
3820 DATA 45,46,50,51,52,53,54,55,2,32,39,40,0,1,2,6,528
3830 DATA 17,18,19,20,23,24,29,30,31,32,34,39,40,43,47,48,
   494
3840 DATA 49,0,1,17,18,19,23,29,30,31,34,43,0,1,17,23,335
3850 DATA 29,34,43,47,0,1,17,23,29,34,43,47,0,1,2,17,367
3860 DATA 20,23,24,29,31,32,34,39,40,43,47,49,3,4,5,7,430
3870 DATA 8,9,11,12,30,48,0,1,2,6,17,18,19,20,23,24,248
3880 DATA 27,28,29,30,31,32,34,39,40,43,47,48,49,0,1,2,480
3890 DATA 17,20,23,24,29,31,32,34,39,40,43,47,0,1,17,23,420
3900 DATA 29,30,34,43,47,27,56,32,36,1,32,35,36,1,32,40,511
3910 DATA 36,1,44,88,41,32,40,36,1,41,44,89,32,36,1,44,606
3920 DATA 88,32,36,1,44,89,32,36,2,1,32,36,2,1,44,88,564
3930 DATA 32,36,2,1,44,89,32,40,36,2,1,41,32,36,3,32,459
3940 DATA 48,42,36,32,27,54,22,15,8,4,1,0,0,0,0,151,440
3950 DATA 150,141,126,103,101,93,77,69,61,50,29,0,0,0,0,65,
   1065
3960 DATA 65,65,66,66,66,66,66,66,66,66,66,66,67,67,67,67,
   1058
3970 DATA 67,67,67,68,68,68,69,73,73,73,74,74,76,76,76,76,
   1145
3980 DATA 78,79,80,80,80,80,82,82,82,82,83,83,83,83,83,83,
   1303
3990 DATA 83,84,84,84,84,84,84,63,68,78,83,67,67,69,73,77,
   1232
4000 DATA 78,80,82,86,86,76,76,76,76,77,80,80,69,69,69,79,
   1239
4010 DATA 78,78,78,77,83,68,68,68,83,79,82,72,72,76,76,79,
   1217
4020 DATA 79,84,84,66,69,69,69,84,84,84,65,65,83,88,88,89,
   1250
```

Before we examine the parameters we will take a look at what relocation is. The following two line example is one that would (or should) never appear but illustrates our relocate command very well.

Location Instruction

C000 LDA#00 ; load accumulator with 0
C002 JMP C000; jump back to C000

Now, if we move this program to some other location in memory, say 8000 hex, we would have the following:

8000 LDA#00
8002 JMP C000

As you can see the loop no longer exists and the program would still jump to C000.

The relocate command will alter any program so that any absolute instructions in the memory range aaaa to bbbb, which refer to any addresses between cccc and dddd, will change any absolute instructions so that they have an offset to eeee. So, to relocate the above example, we would enter the command:

R C000 C004 C000 C004 8000

The program would then be stored in memory in such a way that it would run to 8000. You will have to move the program yourself with the B command to be described later.

Block Move — B

As mentioned above it is possible to move a block of memory from one location to another. The format for this command is:

B ssss eeee nnnn

where ssss and eeee are the start and end of the block which you want to move and nnnn is the new address to which you want to move it. For example:

B A000 BFFF 8000

would move all the memory between A000 and BFFF to memory location 8000 onwards i.e. the Basic ROM would be copied to location 8000 onwards.

Go — G

The G instruction has the following format:

G xxxx

where xxxx is the address from which you wish to run your machine code. It is similar to the Basic command 'RUN linenumber'. However, with 'G', you must enter the execution address.

If, when you run a program, there is a break instruction (i.e. 00) all of the registers will be displayed and you will be returned to the monitor. This is a very handy way of debugging a machine code program as you can force the program to stop wherever you want by inserting 'BRK' instructions.

Find text — F

The 'F' command is used to search through memory for a specific piece of text. Text can either be a character string in quotes, e.g. "PRINT", or a hex number, for example 20, which is the hex code for space. The 'F' command takes the following format:

F ssss eeee text

where ssss is the address where you want to start the search, eeee is the address where you wish to end the search and text is the string that you want to look for. For example the command:

F A000 BFFF "PRINT"

would search through the Basic ROM for every occurence of the word "PRINT".

Put — P

The Put command is used to put a value into every memory location between two set limits. P has the following format:

P ssss eeee xx

Where ssss and eeee are the start and end address and xx is the number with which you wish to fill the memory.

Load — L

It is possible to LOAD a previously saved program back into memory. The command takes the following format:

L text;dd

The command loads the file text from device number dd, e.g. one for tape and eight for disk. The semicolon is very important and should not be omitted. The program will be loaded back into memory at the same location from which it was saved.

Save — S

It is possible to save a block of memory to a specified device with the S command. Save has the following format:

S text;dd,ssss,eeee

where text is the filename, dd is the device number and ssss and eeee are the start and end address + 1 for the save. Make sure you remember to add one to the end address or the last byte of your program will not be saved.

Verify — V

It is possible to verify a program that you have saved with the 'V' command. All you have to do is enter:

V text;dd

where text is the filename and dd is the device number on which the program to be verified is saved.

Decimal to Hex —

This is a very handy command that will allow you to find the equivalent in hex of any decimal number. The format for the command is:

integer decimal value

for example # 10 will print out the hex number 0A.

Hex to Decimal — $

Going from hex back to decimal is also made very easy. All you have to enter is:

$hex number

and the decimal equivalent will be displayed.

Disk Access — >

You are able to send commands to the disk drive by using the '>' command. Any disk command can follow this instruction. For example >"$" will display the disk directory and >"I" will initialise the disk. Do not omit the quotes.

Text Entry — T

With Topmon you can not only store hex numbers into memory but you can also save text strings with the 'T' command. The command has the following format:

T ssss text

where ssss is the address where you want the string to start and text is the string. For example the command:

T C000 "HELLO"

will store the letters H,E,L,L and O from location C000 onwards.

Hunt memory — H

The 'H' command will allow you to search through memory for any references to a block of memory. The command has the following format:

H ssss eeee aaaa bbbb

and will hunt from ssss to eeee for any references to locations aaaa to bbbb.

Disassemble — D

It is not only possible to list the numbers between two locations but it is also possible to print a 6502 mnemonic listing of a block of memory. This makes a program much easier to follow than if it was displayed just in numerical format. The command takes the form:

D ssss eeee

where ssss and eeee are the start and end address for the disassembly.

Printer — *

This is a very simple command that performs a very useful

```
4030 DATA 63,67,68,76,67,83,81,84,73,69,76,75,67,83,67,68,
   1167
4040 DATA 73,86,80,88,89,67,88,89,82,67,88,89,80,82,65,88,
   1301
4050 DATA 89,82,80,65,65,80,65,80,76,82,73,83,67,67,68,73,
   1195
4060 DATA 65,88,89,88,89,88,65,83,65,63,32,201,203,72,32,
   210,1533
4070 DATA 255,173,235,200,240,13,169,3,133,154,104,72,32,
   210,255,169,2417
4080 DATA 4,133,154,32,213,203,104,96,72,165,1,141,255,192,
   9,3,1777
4090 DATA 133,1,104,96,173,255,192,133,1,96,32,201,203,32,
   205,189,2046
4100 DATA 76,213,203,32,201,203,32,138,173,201,145,144,3,
   76,110,194,2144
4110 DATA 32,247,183,76,213,203,32,201,203,162,250,154,76,
   116,164,169,2481
4120 DATA 2,32,195,255,169,3,32,195,255,32,204,255,32,140,
   195,169,2165
4130 DATA 3,162,8,160,15,32,186,255,165,80,162,9,160,192,
   32,189,1810
4140 DATA 255,32,192,255,173,9,192,201,36,240,45,162,3,32,
   198,255,2280
4150 DATA 234,234,234,32,20,195,32,50,195,162,3,32,198,255,
   32,228,2136
4160 DATA 255,32,171,203,201,13,208,246,169,3,32,195,255,
   169,2,32,2186
4170 DATA 195,255,32,204,255,76,170,193,169,2,162,8,160,0,
   32,186,2099
4180 DATA 255,32,192,255,162,2,32,198,255,32,20,195,32,104,
   196,32,1994
4190 DATA 228,255,32,228,255,32,228,255,32,228,255,32,228,
   255,133,76,2752
4200 DATA 32,228,255,166,76,160,128,132,14,32,219,203,32,
   24,195,165,2061
4210 DATA 211,201,4,208,247,32,228,255,166,144,208,64,201,
   34,208,245,2656
4220 DATA 32,171,203,32,228,255,32,171,203,201,34,208,246,
   32,24,195,2267
4230 DATA 165,211,201,23,208,247,32,228,255,201,32,240,249,
   32,171,203,2698
4240 DATA 32,228,255,32,171,203,208,248,32,20,195,165,144,
   72,169,0,2174
4250 DATA 133,153,32,4,197,169,8,133,153,104,240,153,162,0,
   189,237,2067
4260 DATA 204,32,171,203,232,224,13,208,245,76,49,204,66,
   76,79,67,2149
4270 DATA 75,83,32,70,82,69,69,13,0,0,0,0,0,0,0,76,569
4280 DATA 10,69,16,117,11,252,9,246,9,177,10,240,9,225,10,
   204,1614
4290 DATA 20,213,20,228,9,245,12,40,10,52,10,58,10,62,10,
   92,1091
4300 DATA 10,95,10,101,10,125,10,150,10,187,12,204,12,217,
   12,60,1225
4310 DATA 14,64,14,68,14,71,14,74,14,91,14,142,14,156,14,
   164,942
4320 DATA 15,187,17,192,17,211,17,223,17,21,18,28,21,37,21,
   0,1042
```

```
4330 DATA 0,197,10,29,14,192,14,233,14,81,16,11,10,86,21,0,
   928
4340 DATA 0,238,20,137,12,201,12,237,12,101,18,236,18,244,
   18,45,1549
4350 DATA 11,45,11,41,11,43,11,83,10,98,10,117,10,128,10,
   155,794
4360 DATA 10,205,10,211,10,216,10,48,11,52,11,128,11,132,
   11,206,1282
4370 DATA 11,215,11,242,11,63,12,68,12,89,12,92,12,97,12,
   36,995
4380 DATA 13,90,13,141,13,243,13,34,14,39,14,220,14,239,14,
   244,1358
4390 DATA 14,137,15,227,15,3,16,8,16,116,16,121,16,8,16,
   116,860
4400 DATA 16,121,16,208,17,46,13,0,0,130,12,4,18,50,18,40,
   709
4410 DATA 18,43,18,57,18,104,18,146,18,181,15,120,10,4,12,
   62,844
4420 DATA 13,136,13,122,14,181,15,131,10,47,12,93,13,236,
   13,236,1285
4430 DATA 14,92,16,140,10,60,12,100,13,57,14,108,15,98,16,
   167,932
4440 DATA 10,234,12,113,13,99,14,134,15,126,16,249,9,0,13,
   116,1173
4450 DATA 13,109,14,154,15,184,17,55,11,3,13,119,13,112,14,
   157,1003
4460 DATA 15,195,17,254,11,6,13,126,13,115,14,174,15,218,
   17,16,1219
4470 DATA 18,13,21,52,21,55,21,106,21,141,21,174,21,201,21,
   0,907
4480 DATA 0,1,11,11,11,13,11,19,11,31,11,108,10,13,11,215,
   487
4490 DATA 14,135,11,222,11,18,13,159,13,109,21,0,0,29,11,
   54,820
4500 DATA 18,3,19,9,19,5,11,7,11,125,11,129,13,185,13,106,
   684
4510 DATA 14,165,14,168,14,173,14,187,14,13,15,22,15,102,
   15,148,1093
4520 DATA 15,245,16,251,16,240,16,211,21,0,0,9,11,15,11,17,
   1094
4530 DATA 11,23,11,25,11,0,0,17,10,255,10,3,11,33,11,3,434
4540 DATA 11,33,11,0,0,21,11,35,11,37,11,95,16,101,16,137,
   546
4550 DATA 16,238,17,241,17,66,18,178,20,0,0,39,11,12,19,
   246,1138
4560 DATA 17,0,0,92,18,153,18,156,18,166,18,0,0,128,18,134,
   936
4570 DATA 18,140,18,86,10,159,10,17,12,26,12,29,12,33,12,
   110,704
4580 DATA 13,150,13,233,13,226,14,246,15,226,17,24,18,120,
   18,131,1477
4590 DATA 18,137,18,143,18,250,18,171,20,219,20,225,20,228,
   20,244,1769
4600 DATA 20,188,15,238,15,196,20,27,11,247,20,66,21,138,
   21,161,1404
4610 DATA 21,167,21,190,21,196,21,226,21,0,0,41,11,223,21,
   29,1209
4620 DATA 14,234,21,0,0,0,0,216,229,5,83,115,180,228,20,69,
   1414
```

function. It turns on/off any printer connected to the serial bus (i.e. like a Commodore printer) and allows you to keep a permanent record of everything you do.

Back to Basic — X

The command 'X' simply exits the monitor and places you back in basic. The only way to re-enter the monitor is through a SYS command to be described later.

Running The Program

When you LOAD Topmon from tape or disk and RUN it, you will be asked for the page where you would like the monitor to be placed. Topmon is fully relocatable which means that you can place it in any free memory location and it will work correctly. This is very handy when you are writing programs as you can place the monitor where it will not affect what you write.

The memory in your computer is split up into pages. One page being 256 bytes of memory. You can therefore work out what memory any page has by using:

page number * 256

If you want to sit the monitor at a specific location you can work this out with the following bit of maths:

page = location/256

Simple isn't it?

Typing It All In

Topmon is a little complicated to type in so care must be taken to follow the instructions. When you first load Topmon it sits at the same place in memory as would a normal Basic loader. This means that if the loader for the program was RUN, Topmon would write over the program and the machine would more than likely hang. Therefore we have to make the computer think that a Basic program is somewhere else in memory. LOAD your program here, leaving the normal Basic memory free, and run the program. Then all you have to do is SAVE Topmon so that you don't have to go through the loader each time you want to use it.

The first thing to do is type in 'Topmon Load' normally and save this to tape or disk. **Do not run it** or you will have wasted all of your typing.

Next turn off and on your C64 and enter the following line:

POKE 43,0:POKE44,32:NEW

and press RETURN. This will make the computer think that Basic now starts at location 2000 hex. Remember that locations 43 and 44 hold the start of Basic.

Now LOAD in 'Topmon load' with the command:

LOAD "TOPMON LOAD",8
— for disk or

LOAD "TOPMON LOAD",1
for tape.

then RUN the program and Topmon will be moved into the correct area of memory.

All you have to do now is save Topmon on to tape. Enter the following line into the computer:

POKE43,1:POKE44,8:POKE45,40:POKE46,24:CLR

and press RETURN. Make sure you type the line in exactly as shown. Now, if you type LIST, you should see the name Topmon appear. If all is well, then you only need to type:

SAVE "TOPMON",8 for disk
or
SAVE "TOPMON", 1 for tape

and you will have a copy of the program.

Whenever you want to use Topmon in future then just follow the above instructions for running the program.

Errors

If you have made any typing mistakes you will be told when you RUN the 'TOP LOAD' program. Just make the corrections using the normal C64 screen editor and RUN the program again.

Entering the program may sound complicated but is really extremely easy and you should have no problems if you follow the instructions carefully.

MEMORY JUGGLING

David Ford explains how you can get more from Basic memory.

SOMETIMES THE C64'S memory (elephantine though it may be) is just not big enough for that mammoth Basic program – perhaps a never ending adventure game. Or, possibly you just want to have two programs in memory sharing the same data but you dont want to save and reload data and program.

This article describes a technique for the Basic programmer which allows for these possibilities without having to resort to machine language.

How?

The technique is to load your main program into memory at a higher address than usual, and to load a secondary Basic program into the usual Basic area. The two programs can transfer control to each other as required and share the same Basic variables. The secondary program can be changed by loading different secondary programs under the control of the main program, again preserving variables. It sounds pretty easy, but success depends upon following particular rules in a rather strict fashion.

At the heart of the technique is the manipulation of the Basic pointers in zero page (addresses 0 to 255) of the 64's memory. Two key pointers are those that indicate the addresses of the start and end

LISTING No.1

```
10 REM                              ********** loader programme **********
20 :
30 PRINT [cls]                        : REM ***** clear screen
40 POKE 251,PEEK (53281)              : REM ***** save screen colour
50 POKE 53281,PEEK (53280)            : REM ***** blank screen to border colour
60 POKE 252,PEEK (646)                : REM ***** save ink colour
70 POKE 646,PEEK (53280)              : REM ***** set ink to border colour
80 POKE 631,76                        : REM ***** ASCII 'L' into buffer
90 POKE 632,207                       : REM ***** <SHIFT>'O' into buffer + 1
100 POKE 633,13                       : REM ***** <RETURN> into buffer + 2
110 POKE 634,82                       : REM ***** 'R' into buffer + 3
120 POKE 635,213                      : REM ***** <SHIFT>'U' into buffer + 4
130 POKE 636,13                       : REM ***** <RETURN> into buffer + 5
140 POKE 198,6                        : REM ***** set length of buffer queue
150 :
160 POKE 16384,0 : POKE 44,64 : NEW : REM ***** move BASIC set  pointers,
and terminate.
170 END
```

LISTING No.2

```
5 REM                               ********** main module **********
7 :
10 POKE 53281,PEEK (251)              : REM ***** restore screen colour
20 POKE 646,PEEK (252)                : REM ***** restore ink colour
30 POKE 44,8 : LOAD                   : REM ***** move BASIC and LOAD programme
40 :
50 REM                              ********** main prog starts here **********
60 :
100 ROOM = 1
110 POKE 44,8 : POKE 58,255 : ON ROOM GOTO 100,200,300
120 PRINT "[rev]F[off]ORWARD OR [rev]B[off]ACKWARD ?[2cd]"
130 GET Z$ : IF Z$ <> "F" AND Z$ <> "B" THEN 130
140 IF Z$ = "F" AND ROOM < 3 THEN ROOM = ROOM + 1 : GOTO 110
150 IF Z$ = "B" AND ROOM > 1 THEN ROOM = ROOM - 1 : GOTO 110
160 IF Z$ = "F" AND ROOM = 3 THEN POKE 44,8 : LOAD
170 GOTO 110
180 END
```

of a Basic program. They are stored at addresses 43/44 for the start of program pointer and 45/46 for the end of program pointer. For those unfamiliar with pointers, I'm including a short description.

Pointers

A pointer is a location in memory which contains another memory address between zero and 65535. Nearly all the addresses in the 64 are greater than 255 and, since one byte can only hold a number up to 255, two bytes must be used to store an address. These two bytes are called the low byte and the high byte and the former always precedes the latter in memory. The low byte is actually a straightforward counter going from zero to 255. When address 256 is reached the low byte returns to zero and the high byte is set to one. Counting would then continue in the low byte up to address 512 (2 * 256) when the low byte would again be set to zero and the high byte would be incremented to two. So any address is given by:

PEEK (low byte) + 256 * PEEK (high byte)

At switch-on, the values in addresses 43 and 44 (the low and high bytes of the start of Basic) are one and eight. Therefore a Basic program would start at 1 + (256 * 8) or address 2049. Actually, the start of the Basic programming area is one byte below this at 2048 and this byte is always set to zero to mark the start of the area available to Basic. The Basic program area actually extends from 2048 up to 40959 (nearly 38K) and within reason a program can be placed anywhere within this area.

On The Move

For these examples Basic will be moved so that it starts at 16384 (some 14K further up the memory map than usual). This is achieved in direct mode by two POKEs and a NEW.

POKE 16384,0 : POKE 44,64 : NEW

The first POKE to 16384 marks the start of the new programming area in just the same way as the byte at 2048 is also set at zero. The POKE to 44 changes the actual pointer to the start of the Basic program, and the NEW command automatically updates all the other pointers used by Basic. Only the high byte of the pointer has been altered because Basic has been moved by an exact multiple of 256, so the low byte is unaffected. Moving Basic in multiples of 256 makes the manipulation of pointers easier when programming. The new address for the start of a Basic program is now 1 + (256 * 64) or 16385. If you were now to type a program or LOAD an existing program from tape, it would start at 16385 and there would be 14K of free space under the Basic program for sprite data, user defined characters or a high resolution screen.

In Style

This is all very fine, but rather clumsy. Surely all this POKEing and LOADing could be done under program control? Yes it could but there are problems. Typing a program to carry out the above work, for example;

10 POKE 16384,0 : POKE 44,64 : NEW : LOAD

will not work because the program will be removed from memory by the NEW before it gets to LOAD. Also, the end of program pointer (45/46) will not be resest by a LOAD instruction issued from within a program, so Basic will get totally confused.

Fortunately, there is a way around these problems. By POKEing instructions into the keyboard buffer, the 64 can be instructed to carry out an operation immediately it exits from Basic – after a NEW for example. Whilst such an instruction seems (to the user) to be under program control, the 64 regards it as having been issued in direct mode and therefore sets all its pointers correctly. Listing 1 is a loader program which achieves this effect in a little more style than POKEing in direct mode.

Lines 30 to 70 are purely cosmetic. Because the LOAD instruction is issued in direct mode, the tape will stop and the screen will display 'FOUND PROGRAM' as normal. This looks a little amateurish, so the screen is blanked to simulate one continuous load. The tape will still pause for a few seconds, but nothing will show on the screen. From line 80 onwards, abbreviated keywords are POKEd into the keyboard buffer. The abbreviated keywords have been used to save space, and the buffer contains the instructions 'LOAD' and 'RUN' to automatically load and run the next program on the tape. The last line copies the commands used in direct mode. Type Listing 1 now and save it at the beginning of a new tape.

Disk Users Do It Too

Disk users have a slight problem in that the LOAD instructions for disk use are too long to put in the keyboard buffer, yet they perhaps have the most to gain from this technique. However, all is not lost, instead of the keyboard buffer POKEs shown in lines 80 to 140 use the following procedure:

80 NAME$ = "PROGRAM"
90 PRINT "[cls] LOAD NAME$, 8 : RUN"
100 POKE 631,19 : REM cursor home
110 POKE 632,13 : REM return
120 POKE 198,2 : REM No. of characters in buffer

This is a variant of POKEing the buffer, whereby the actual instruction is printed on the screen and the keyboard buffer

LISTING No.3

```
  0 POKE 44,64 : POKE 58,255 : GOTO 100
 10 :
 20 REM                    ********** secondary prog 1 **********
 30
 80 REM                    ***** room 1
 90 :
100 PRINT "YOU ARE AT THE BACK DOOR"
110 GOTO 400
170 :
180 REM                    ***** room 2
190 :
200 PRINT "YOU ARE IN THE HALL"
210 GOTO 400
270 :
280 REM                    ***** room 3
290 :
300 PRINT "YOU ARE AT THE FRONT DOOR"
310 :
400 POKE 44,64 : POKE 58,255 : GOTO 120
410 END
```

simply contains the carriage returns to enter the line.

Sprites, UDGs And Hi-res

In itself Listing 1 will be useful. It is all that is required to relocate Basic and LOAD any program which requires access to large amounts of sprite or character data, or to a high resolution screen. These can all fit into the area from 2048 to 16383 (14K) without fear of corruption by Basic and there is still 24K left for use by a Basic program. Personally, I think that this is a more elegant method for most purposes than resetting the Vic chip to look at another bank of memory. Nevertheless, the purpose of this article is to run two Basic programs concurrently, so, here goes.

Now that Basic starts at 16384, you can LOAD the main program into this space. The main program should, of course, be recorded on the tape immediately after the loader program. The ultimate purpose of the exercise is to have two programs in memory, so the first task of the main program should be to LOAD the secondary program. The first lines of the main program might therefore look something like this:

```
10 POKE 53281,PEEK (251) : REM restore screen colour
20 POKE 646,PEEK (252) : REM restore ink colour
30 POKE 44,8 : LOAD
```

The first two lines restore the screen and ink colours which were changed by the loader program. The third line changes the start of Basic pointer back to 2049 and then LOADs the secondary program starting at the usual address. This LOAD can be given from within a program because there is no NEW command, and **must** be given from within a program because the end of Basic pointers must not be reset after this LOAD. The only constraint is that (in this example) the second program must not exceed 14K in length or it will over-write the main program. If longer secondary programs are to be used then the main programs must start higher than the 16385 used in this example. This is achieved by

LISTING No.4

```
  0 POKE 44,64 : POKE 58,255 : GOTO 100
 10 :
 20 REM                 ********** secondary prog 2 **********
 30
 80 REM                 ***** room 1
 90 :
100 PRINT "YOU ARE IN THE FRONT GARDEN"
110 GOTO 400
170 :
180 REM                 ***** room 2
190 :
200 PRINT "YOU ARE AT THE GATE"
210 GOTO 400
270 :
280 REM                 ***** room 3
290 :
300 PRINT "YOU ARE IN THE STREET"
310 :
400 POKE 44,64 : POKE 58,255 : GOTO 120
410 END
```

POKEing a figure higher than 64 to address 44 in the loader program. Every increment of address 44 moves Basic by 256 bytes.

Back To Base

When the secondary program has LOADed it will automatically be RUN. Note that it will be the secondary program which is RUN and not the main program. Therefore it will normally be necessary for the secondary program immediately to pass control back to the main program. Assuming that the main program proper starts at line 100 (after the three line loading routine shown above), then the first line of the secondary program should look like this:

```
0 POKE 44,64 : POKE 58,255 : GOTO 100
```

The effect of this line is perhaps not self-evident, but it serves the purpose of transferring control between the two programs. It is listed as line zero so that it cannot be inadvertantly preceded by another line in the program.

The first POKE to 44, restores the start of Basic pointer to 16385 (i.e. it points to the main program). The second POKE to 58 sets the current Basic line number to an out of range figure. If this were not done in a jump from a lower (0) to a higher (100) line number then Basic would continue searching for line 100 in the secondary program with disastrous results. POKEing 58 with 255 fools the 64 into thinking that it is on a Basic line with a number of about 65000 – an illegal figure – and so it must start to search from the beginning of the program to find line 100, and the beginning has just been set to 16385 – the main program. The last instruction is obvious – GOTO line 100 – but in the main program not the current one.

Into Action

Now that the basic concept has been introduced, it is time to see it in action. Type in Listing 2 – the main program – and SAVE it to tape immediately after the loader program. Then NEW and type in Listing 3, SAVEing it to tape immediately after Listing 2. Finally, NEW and type Listing 4 SAVEing that after Listing 3. Got that OK? Before you LOAD the whole package let's have a look at what will happen.

Firstly, the loader program will LOAD, change the Basic pointers and LOAD the main program at 16385. The main program will then RUN, changing the Basic pointer back to 2049 and LOAD the first of the secondary programs. The secondary program will then autoRUN and immediately pass control back to the main program at line 100. At this point the rather simple example begins. This is a crude module permitting movement between three locations or rooms. At the start the room number is set to one. Line 110 changes the start of Basic pointer in 43/44 to point to the secondary program and then passes control to that program at either lines 100, 200 or 300 depending on the room number. The secondary program then prints the relevant room description and, after changing the start of Basic pointer back to the main program at 16385, returns

control to the main program at line 120.

The remaining lines allow you to choose between going backward or forward. When room three is reached the command to go forward causes the program to LOAD the next secondary module which interacts with the main program in exactly the same way as the first, but has new desriptions for each location. If you go past room three in this program it will attempt to load a further module from tape which does not exist. These programs are very simplistic but serve to make the point. Disk-drive systems present more scope than tape only systems because they can go back and forth between programs, whilst tape systems can only go forward.

Load And Run

Rewind the tape now and LOAD the program from the start using RUN/STOP. If the tape motor stops during the process but the screen remains blank, don't worry – either wait for 10 seconds, or if you're impatient, press the Commodore key. The loading process will end and you will be informed that 'You are at the back door'. If you step forward and pass through the front door, then the second module will LOAD (after pressing play) and you can then continue out into the street.

Last Thoughts

That's about all there is to it, but if you want to use this technique there are just a few final points:

a) If you break into the program, you can examine either listing by using POKE 44,8 : LIST to see the secondary program and POKE 44,64 : LIST to see the main program.

b) Don't try to edit a secondary program at 2049 with the main program in memory at 16385. You will corrupt the main program and the pointers won't be set properly to SAVE the corrected secondary program anyway.

c) Control can be transferred between programs using GOTO, ON GOTO, GOSUB and RETURN provided that each command is preceded **on the same line** by POKEing locations 44 and 58.

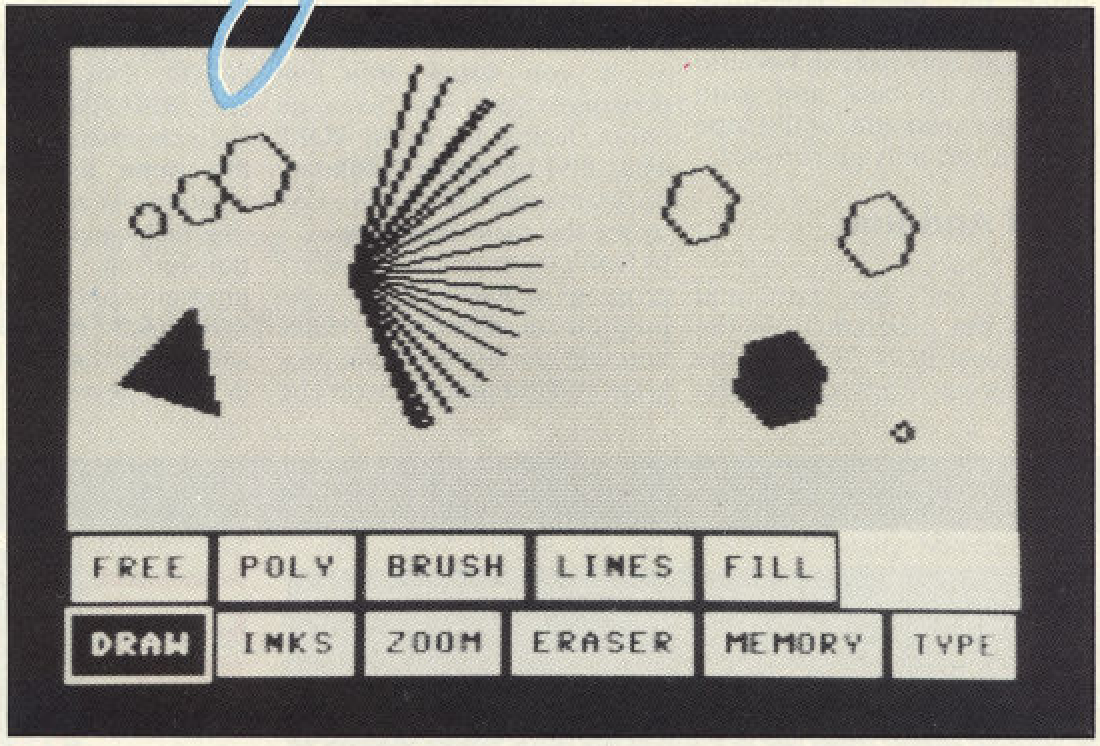
d) DATA can be READ without transferring control to the program containing the DATA. POKE 66,8 : RESTORE : READ will READ the first DATA item in the secondary program, whilst POKE 66,54 : RESTORE : READ will start READing the first DATA item in the main program.

Just try it out and use your own ideas. You should find it quite a useful technique, and not a bit of machine code in sight! Perhaps someone could start writing a never ending adventure program in instalments (Coronation Street on computer?), if they could produce new secondary modules faster than people could solve them!

MEMORY JUGGLING

MAKING light WORK

Eric Doyle's artistic talent was tested to the full with Amicron's light pen.



IN THE COMPUTER WORLD FEW ACTivities give you a greater sense of interaction than the uses of a lightpen. To actually control the cursor on the screen, from the screen, has always seemed like magic to me and Amicron's lightpen provides a very smart magic wand in a neat little package.

The pen is robust and yet compact, being no bigger than a normal upmarket ballpoint pen. The shaft is made of stainless steel with a removable plastic tip which switches on the pen when pressed against the TV screen. The tip can easily be pulled out for cleaning and this does make me a little concerned that, if the tip were to be lost, the pen would be unusable. I've tried shaking the pen around and as yet the tip has remained firmly in position, so perhaps I shouldn't worry too much.

The pen is connected to the joystick port of the C64 by a coiled lead which stretches to about two metres. This type of cable looks very neat and practical, until you want to use it. Suddenly you find your precious computer has been dragged to the floor as you try to overextend the lead. I suppose the alternative would be equally unattractive, having two metres of cable lying around amongst all the other necessary cables and connectors.

The pen also includes a software tape and a booklet which is all you need for constant experimenting. The tape has the expected graphics program but also includes an application demonstration called Pencalc. This draws a calculator on screen and you can use the lightpen to operate its keyboard.

Additionally, the tape has a program called Pencode which can be used in your own programs to control the lightpen's operation. This is because the normal PEEKs and POKEs recommended by Commodore need smoothing out a little.

Lightpens work because the computer sends its own signal to trigger the tiny beam of light which draws, or scans, your TV picture. The computer 'knows' how long this takes and can calculate where the scan is at any time during the fraction of a second which it takes to cover the screen, line by line. The lightpen pulses a signal to the computer when the scanning beam passes its tip and the computer can then calculate the X and Y coordinates of the position of the pen.

The Graffiti graphics program is fun to use but, like most graphics packages, it is an end in itself rather than being a programming aid. The drawings produced can be saved to tape or disk but there are no instructions on how these designs can be loaded for use in your own programs.

As a demonstration, Graffiti is an example of icon and lightpen driven software. There is room for improvements such as the inclusion of BOX and CIRCLE commands though the overall design using menus calling sub-menus ensures that the screen remains childishly easy to operate.

The Pencode program details also include instructions for reading the pen co-ordinates. One very useful feature is the ability to call up sprite zero to automatically shadow the tip of the lightpen with a user defined shape.

Using the pen is very simple but like all lightpens I find it to be slightly wayward if pixel point accuracy is required. This is not altogether the pen's fault but has more to do with the difference between the computer's perfect concept of a frame scan and the practical reality.

As a cheap, reliable utility, the lightpen functions perfectly well but I think I would have preferred more thought in the preparation of the software. I admit that Amicron, through Mirrorsoft, are only marketing a lightpen and that the software is a secondary consideration but for many customers it is the software that would be the greatest selling point.

Touch Line
Name: Microscribe Lightpen
Machine: C64
Price: £29.95
Distributor: Mirrorsoft, Purnell Book Centre, Pulton, Bristol BS18 5LQ.

22 FOR THE C128

Steve Carrie brings us our first utilities for the 128.

THE TWO PROGRAMS GIVEN here are intended to run on a C128 system with at least one disk drive. They will work with 1541, 1570 or 1571 disk units.

The first program is a utility for setting-up the so-called "boot-sector" on a disk to allow auto-booting by the 128. The second program makes use of this facility to set automatically the 128 to C64 mode and load a program.

Type in both programs and save them to disk. Note, in Program 2, the REM statements detailing the ASCII code of the first letter of the C64 file to be loaded. If you have a program on your C64 disk which you wish to load every time you power-up, then change this to the ASCII code of the first letter of that filename.

Setting-up Your Boot Disk

The process is simple.
1. RUN Program 1. You will be asked to place your disk in the drive and press the return key when you are ready. This disk will be referred to subsequently as the boot disk. You will then be asked if you want to format the disk. If you answer "Y" you will be asked for the name of the disk and the ID. If you give only the disk name (and press only the return key when asked for the ID) a directory clear-out will be performed. Otherwise a complete format operation will follow. Next, the program makes sure that the disk boot sector is free. If not, the program will stop indicating that some other program is using this sector. If all is well you will be asked for the name and type of the file you wish to autoboot. The file type is either Basic or Binary (machine code). Note here that to set up the C64 autoboot program, you must give the filename used in the BSAVE command in Program 2 and the file type must be Binary. When you have selected the file type, the program performs the set-up.
2. LOAD Program 2 and make sure that the filename in the BSAVE command is correct and check that the first letter of the C64 filename is correct.
3. Place your boot disk in the drive and RUN Program 2. The machine code file will be saved to disk.
4. If you already have a program on the disk which will load in C64 mode, then you can try out the disk by simply pressing the reset button. If you don't have a program ready, type in the small demo program below:

```
10 PRINT "[CLR] NOW IN C64 MODE"
20 PRINT "[DOWN][DOWN] YOUR COMMODORE 1986"
```

Save this with a filename beginning with the letter you selected in Program 2. Now press the reset button. The following should happen:
1. The usual 128 title screen appears.
2. The message BOOTING < filename > appears.
3. The machine enters C64 mode.
4. The program loads and runs.

Program 2 has checksum error detection facilities which will detect any error in the numeric values contained in the DATA lines. Since one error may cancel out another, be sure to check your typing.

The C64 Autoboot Program (program 2)

Program 2 works by changing the memory configuration to that used by C64 mode and then simulating the C64 ROM cold start routines. Finally, it fills the keyboard buffer with the characters to load and run the C64 program.

The program resides in bank zero RAM memory at address 3000 hex. Since this is the bank used by C64 mode. Any machine code programs in the area 0802-CFFF hex are unaffected by the mode change.

What follows is basically an inspired guess since no programmer's reference guide is yet available (Commodore please note!).

The Memory Management Unit (MMU) resides in the I/O space at D500 hex with the first five registers (D500-D504 hex) also appearing at address FF00 in all banks. Most of the system software addresses the MMU at FF00. As far as I can gather from

PROGRAM 1

```
10 trap 540
20 color 0,2:color 4,4:n$=chr$(0)
30 print chr$(14);"[CLEAR]  [BLUE][RVSON]Commodore 128 Disk Boot Sector Setup"
40 print "[DOWN][DOWN][DOWN] Please insert your new disk in drive 8 and press return when ready"
50 do:getkey a$:loop until a$=chr$(13)
60 print "[DOWN][DOWN][DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]Format disk (y/n)?"
70 do:getkey a$:loop until a$="y" or a$="n"
80 ifa$="y" then begin
90 input "[DOWN][RIGHT][RIGHT]Disk name";dn$
100 id$="":input "[DOWN][RIGHT][RIGHT]Disk ID";id$
110 if id$<>"" then print "[DOWN][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]Full format operation selected":else print"[DOWN][RIGHT][RIGHT][RIGHT][RIGHT]Directory clear-out selected."
```

```
120 ac$="n0:"+dn$+","+id$
130 open15,8,15,ac$:close15
140 if ds then print "[DOWN][DOWN]
[RIGHT][RIGHT][RIGHT][RIGHT][RED][
RVSON]DISK ERROR DETECTED... abort
ing.[BLUE]":end
150 print "[DOWN][DOWN][RIGHT][RIG
HT][RIGHT][RIGHT][RIGHT][RIGHT][RI
GHT][RIGHT][RIGHT][RIGHT][RIGHT]Fo
rmat complete."
160 sleep 2
170 bend
180 open 15,8,15,"i":open2,8,2,"#"
190 print#15,"b-a 0 1 0"
200 if ds then begin
210 close 2:close 15
220 print chr$(7);"[DOWN][DOWN][RE
D][RVSON][RIGHT][RIGHT][RIGHT][RIG
HT]Boot sector is in use. Aborting
..[BLUE]"
230 end
240 bend
250 do
260 scnclr
270 print "[DOWN][DOWN][RIGHT][PUR
PLE]Please enter the name of the p
rogram     you wish to autoboot."
280 print "[RIGHT](up to 16 charac
ters in length)
290 input"[DOWN][DOWN]";f$
300 if len(f$) >16 then begin
310 print chr$(7);"[DOWN][DOWN][RI
GHT][RIGHT][RIGHT][RED][RVSON]File
name too long. Please re-enter"
320 sleep 2
330 bend
340 loop until f$<>"" and len(f$)<
17
350 print "[CLEAR][DOWN][DOWN][RIG
HT][RIGHT]Is your program..[DOWN][
DOWN]"
360 print "[RIGHT][RIGHT]1. Basic"
370 print "[RIGHT][RIGHT]2. Binary
 (i.e. machine code)"
380 print "[DOWN][RIGHT][RIGHT]Sel
ect (1) or (2) now."
390 do:getkey a$:loop until a$="1"
 or a$="2"
400 print"[DOWN][DOWN][RIGHT][RIGH
T][RIGHT][RIGHT][RIGHT][RIGHT][RIG
HT][RIGHT]Setting up boot sector."
410 if a$="1" then m$="run":else m
$="boot"
420 i$=m$+chr$(34)+f$+n$
430 v0=len(f$)+2831:vl=(v0 and 255
):vh=int(v0/256)
440 print#15,"u1:2 0 1 0"
450 print#15,"b-p 2 0"
460 b$="cbm"+n$+n$+n$+n$+f$+n$+n$
470 print#2,b$;
480 print#2,chr$(162);chr$(vl);chr
$(160);chr$(vh);chr$(76);chr$(165)
;chr$(175);
490 print#2,i$
500 print#15,"u2:2 0 1 0"
510 close2: close15
520 print "[DOWN][DOWN][DOWN][DOWN
][DOWN][RIGHT][RIGHT][RIGHT][RIGHT
][RIGHT]Disk boot sector setup com
plete."
530 end
540 rem ***************************
***
550 rem error trap routine
560 rem ***************************
***
570 dclear
580 print chr$(7);"[DOWN][DOWN][RE
D][RVSON][RIGHT][RIGHT][RIGHT]Prog
ram operation error..Aborting.[BLU
E]"
590 end
```

long sessions of "hacking" about, address FF00 is the memory configuration control register. Addresses FF01 to FF04 (apparently read only – but read on) seem to act as preset configuration switches/latches which, when stored to (e.g. STA $FF01) set FF00 to the value contained in them. For example, FF01 normally contains the MMU value for Bank zero (3F hex). The instruction STA FF01 regardless of the contents of the accumulator will change FF00 to the value in FF01 (3F hex). These preset values may be changed by addressing locations D501 to D504 but since the system is always running, I don't recommend it!

The preset values under normal conditions are:

Addresses		Value	Configuration
FF01	(D501)	3F	Bank 0
FF02	(D502)	7F	Bank 1
FF03	(D503)	01	Bank 14
FF04	(D504)	41	an odd one this; seems to be a mixture of banks 14 and 5 possibly allowing external ROMs to be addressed.

The values in FF00 for each of the 16 pre-defined banks are (starting at bank 0):

bank 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
value 3F 7F BF FF 16 56 96 D6 2A 6A AA EA 06 0A 01 00

How these affect the memory configuration is shown in the C128 system guide on page 17-5.

Memory addresses 0000-03FF hex seem to be common to all configurations hence the Basic and Kernal subroutines are here.

I hope some of you find these programs and my guesswork on the C128 system useful.

PROGRAM 2

```
10 bank0:s=12288
20 for ln=100 to 210 step 10
30 tt=0:t=0
40 for by=1 to 8:read a:pokes,a:s=
s+1:t=t+a:next
50 read tt:if t<>tt then print"dat
a error in line";ln:end
60 next
70 print "[DOWN]m/c installed. sav
ing to disk"
80 bsave "autoboot",u8,b0,p12288 t
o p12384
90 print "finished.":end
100 data 165,186,9,48,141,88,48,16
9,854
110 data 227,162,47,134,0,133,1,16
2,866
120 data 0,169,247,142,0,255,141,5
,959
130 data 213,142,48,208,162,255,12
0,154,1302
140 data 216,169,0,141,22,208,32,1
63,951
150 data 253,32,80,253,32,21,253,3
2,956
160 data 91,255,88,32,83,228,32,19
1,1000
170 data 227,32,34,228,162,251,154
,162,1250
180 data 0,189,81,48,157,119,2,232
,828
190 data 224,10,208,245,134,198,76
,134,1229
200 :
210 data 227,76,207,34,66,42,34,44
,730
220 rem ------------------^^
230 :
240 rem this is the acsii code of
the
250 rem first letter of the c64 pr
ogram
260 rem which will be loaded (see
text)
270 :
280 data 56,58,131,0,0,0,0,0,245
```

MAIL ORDER ADVERTISING

British Code of Advertising Practice

Advertisements in this publication are required to conform to the British Code of Advertising Practice. In respect of mail order advertisements where money is paid in advance, the code requires advertisers to fulfil orders within 28 days, unless a longer delivery period is stated. Where goods are returned undamaged within seven days,the purchaser's money must be refunded. Please retain proof of postage/despatch, as this may be needed.

Mail Order Protection Scheme

If you order goods from Mail Order advertisements in this magazine and pay by post in advance of delivery, Argus Specialist Publications Ltd will consider you for compensation if the Advertiser should become insolvent or bankrupt, provided:

(1) You have not received the goods or had your money returned; and

(2) You write to the Publisher of this publication, summarising the situation not earlier than 28 days from the day you sent your order and not later than two months from that day.

Please do not wait until the last moment to inform us. When you write, we will tell you how to make your claim and what evidence of payment is required.

We guarantee to meet claims from readers made in accordance with the above procedure as soon as possible after the Advertiser has been declared bankrupt or insolvent (up to a limit of £2,000 per annum for any one Advertiser so affected and up to £6,000 per annum in respect of all insolvent Advertisers. Claims may be paid for higher amounts, or when the above procedure has not been complied with, at the discretion of this publication but we do not guarantee to do so in view of the need to set some limit to this commitment and to learn quickly of readers' difficulties).

This guarantee covers only advance payment sent in direct response to an advertisement in this magazine (not, for example, payment made in response to catalogues etc., received as a result of answering such advertisements). Classified advertisements are excluded.

YOUR COMMODORE

Lineage: 40p per word.
(46p incl. VAT).
Semi display: £9.00 per single column centimetre (plus VAT). Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01-437 0699
EXT. 291

Send your requirements to:
STUART TAYLOR
ASP LTD, 1 GOLDEN SQUARE, LONDON W1

SOFTWARE

KNOCKDOWN PRICES
on
PLUS 4/C16 SOFTWARE
with over
125 TITLES TO CHOOSE FROM
DISCOUNTS FROM 10% - 50%
INCLUDING MANY OF THE LATEST TITLES
TO RECEIVE YOUR FREE LIST
SEND (A2) SAE TO:—
Venturegate Ltd (C16/PLUS4 Software List)
17 Harold Road, Leeds LS6 1PR, W. Yorkshire, England.

BANKER 64

If you are fed up not being able to keep track of your bank account this program may be just what you are looking for. Simple to use as it is menu driven it will hold full details of Credits, Debits, Standing orders and Direct Debits and will automatically place them in date order. It will give you statements to screen or printer including the 1520 plotter printer. Records can be stored on both disk and tape for complete back-up. Entry to any account is by name and Security code which can be changed as necessary. This m/c program is very good value at **£6.25 disk, £4.00 tape.**

RAMFILE

Keep track of your Software, Records, Stamp Collection, Addresses etc. with this easy to use menu driven filling system. Entry titles are user defineable. Powerful options include search and amend routines. Print routines allows printing of entry title, selected entry or complete file to any printer including the 1520 plotter. Files can be stored on both disk and tape.
This m/c program is very good value at **£6.25 disk, £4.00 tape.**

***Special Offer* — Both programs £10.Disk Both programs £7.00 Tape**
All prices include VAT P&P Cheques & P.O. to
RJP ELECTRONICS
27 Denham Way, Camber, Rye, East Sussex TN31 7XP Tel: (0797) 225549

AFFORDABLE SOFTWARE

Chuckie Egg (c)	£6.45
Hacker (c)	£7.95
Fighter Pilot (c)	£7.75
Commando (c)	£7.95
Goonies (c)	£7.75
Solo Flight (c,d)	£10.95

All prices include postage and packing. Send a S.A.E. for price list. To order any of the above titles send a cheque or postal order payable to:-
J + J SOFT SALES (MAIL ORDER)
DEPT CU, 17 VERNON ROAD, UCKFIELD, SUSSEX, TN22 5DX

SENSATIONAL POSTAL software bargains for Amstrad, Atari, BBC, Commodore (all formats), Electron, Spectrum. For free list send a SAE or phone Teletex Cleckheaton 0274 875299. 28 Brooklyn Court, Bradford Road, Cleckheaton, W. Yorks.

FRENCH REVISION
"La Conjugaison"
30 Regular Verbs. 17 Tenses for C.S.E., 'O', 'A' Level. As used in schools.
Commodore 64 Disk **£13.95** + 50p p&p.
ACCESS & CHEQUES WELCOME
Enquiries & Orders to:
MEDSTAT LTD, City House, Maid Marion Way, Nottingham NG1 6BH
Tel: 0602 411120

ALPHASOFT
CBM64 Software - 25% Reductions

Commando **£7.45**	Ace **£7.45**
Monty On The Run **£7.45**	Goonies **£7.45**
Outlaws **£7.45**	Winter Games **£7.45**
Never Ending Story **£7.45**	Ball Blazer **£7.50**
Fight Night **£7.45**	Rambo **£6.90**

AND MANY MORE!
Cheque/P.O. to **ALPHASOFT**
115 Parsonage Road, Withington, Manchester M20 9WZ
(List sent with order, P&P included)

GIVE YOUR BUSINESS A BOOST RING STUART TAYLOR ON 01 437 0699

DISK AID
Menu driven utility inc. file recovery, directory reshuffle, fast format etc. Saves time and hassle with the 1541. **£8.95** (inc. P&P)
BARIM SOFTWARE,
4, Wellington Road, Ashford, Middx TW15 3RJ.

MUSI-SYNTH
Powerful 100% M/C **MUSIC PROGRAM** for C16 & Plus/4. Includes Full ADSR & Tone Envelopes, Glide, Vibrato, Tremelo, etc. Record, Playback & Edit Your Own Tunes in Both Voices.
Full Instruction Manual, including details about using tunes in your own programs. **ALL THIS & MUCH MORE ONLY £6.**
Cheques/P.O. to:
JAFSOFT (DEPT YC)
5 Cissbury Road, Ferring, West Sussex BN12 6QJ

DERBY DAY - C16 Can you pick a winner in this Horse-Racing Spectacular? On Screen Action, visual form guide. Over 50 horses, thousands of race combinations. Realistic betting from different meetings. Plus many more features. Send £5.75. Cheque/PO to **BLOOMFIELD SOFTWARE,** P.O. Box 69, Blackpool FY2 9EX

FRUIT MACHINE for CBM 64, Nudges, Gambles, Holds, etc. £4.95. Cheque/P.O. to R. Brafield, 60 Knightcott Road, Banwell, Avon BS24 6HF.

SOUTH EAST KENT SOFTWARE
Dept Y/C P.O. Box 152 Orpington Kent
We stock for Spectrum, Amstrad, Commodore, Send S.A.E. for full list and prices. Please state which.
Postage/Packing 50p, overseas £1.00.
COMMODORE 64 GAMES (TAPE)

	RRP	Our Price
Commando	£9.95	£7.30
Goonies	£9.95	£6.99
Friday 13th	£8.95	£6.50
Scooby Doo	£7.95	£5.70
Winter Games	£9.95	£7.30
Summer Games II	£9.95	£6.70
Geoff Capes	£8.95	£6.99
Skool Daze	£6.95	£4.95
Transformers	£8.95	£6.99
Never Ending Story	£9.95	£6.50
Rambo	£8.95	£6.99
Fightnight	£9.95	£7.00

BUSINESS (DISK)

	RRP	Our Price
Maillist	£24.95	£18.95
Superbase 64	£99.95	£60.95
Cash Book	£59.95	£48.95

HANDWARE

	RRP	Our Price
Quickshot II	£11.95	£7.95
Commodore 16 Joystick	£9.95	£8.69

Cheques/PO's payable to: **J Ferguson.**
Please allow 21 days delivery

CLAIRES COMPUTERS
TOP Software at **LOWEST** Prices.
Order two titles and you will receive a Top Class game **FREE**

	TAPE	DISK
Fight Night	£7.50	£10.50
Bounder	£6.95	
Revs	£9.95	£12.95
The Eidolon	£7.50	£10.99
Rambo	£7.50	
Commando	£7.50	
Yie Ar Kung Fu	£6.50	
They Sold A Million	£7.50	£10.50
Desert Fox	£7.50	£10.50
Winter Games	£7.50	£10.99
Little Computer People	£7.50	£10.50
Korinis Rift	£7.50	£10.50

Quick Shot II Joysticks Only £6.50
This is only a selection from our stock
Send now for latest list.
Send Cheque's P/O to
Claires Computers, 16 Treganoon Rd, Mount Ambrose, Redruth, Cornwall, TR15 1QZ
Tel 0209 211930 for your software needs

GRAPHICS

PIXEL-PAL
For Commodore 64
- Plan your Graphics
- Simple to use "£4.95 incl. P&P"
- 50 A3 sheets
- Programming Guide

COMPUTER BITS
35 Windermere Drive, Biggleswade, Beds.

BOOKS



We use only 10% of our mental potentials

Einstein

These are the words of Albert Einstein, the greatest physicist of recent times.

L. Ron Hubbard's discoveries in the field of the mind prove today that Einstein was right.

In his book "DIANETICS: The Modern Science of Mental Health" L. Ron Hubbard takes one more giant step in this direction.

He reveals how anyone can use his discoveries to isolate the exact barriers that have so far prevented people from using their mental potentials to the full.

More and more people from all walks of life use Dianetics technology today.

The many written reports on their success are the best proof of the quality of this book.

Find our for yourself. Order your copy of this remarkable book today.

Price £3.95 paperback £17.50 hardback. Make cheques and POs payable to S.H.F. Send to Church of Scientology Saint Hill Foundation, FREEPOST (YC 1), East Grinstead, Sussex RH19 4ZA (no stamp required).

WARNING NOTICE

Advertisements placed in this magazine are to be in strict compliance with our standard conditions (copies of which conditions are available on request) and on the clear understanding that the advertiser warrants that his advertisement(s) does not infringe any copyright or condition of sale of any interested party in the advertised product.

Further, the advertiser indemnifies the proprietors of this magazine in respect of costs, damages, or any other claims brought against them as a result of legal action arising from the publication of the advertisement.

Any breach of these terms or the said conditions may result in prosecution of the advertiser by the proprietors.

FOR SALE

MOONRAKER £7.95
CBM64 Space Game features fast
★Fly-into Rolling Radar Display
★Intra Red Target Screen
Altitude, Time, Score Indicators,
Fuel/Purchase Option, Promotion.
Send your name, address & PO/cheque to:
J & J Gilmour Ltd.,
Dept YC486, 4 Copse Close, Liss,
Hampshire GU33 7EW, England.
Tel: 0730-893430.

SOFTWARE GAMES

BLOCKBREAKER — C64 wordgame includes: Challenge Games, Gold runs, money, prizes. **£4.45** Cheque/PO to Drive, 92 Robin Hood Lane, Birmingham B28 0JX.

FOR HIRE

VIC-20/CBM-64 SOFTWARE HIRE

Free Membership. Hundreds of Titles to choose from. Send S.A.E. for your Free Hire Kit:
VSH (YM), PO BOX 65, NOTTINGHAM NG1

COMMODORE 64 cassette library. Hire latest games and educational software, only one price £1.50 per cassette 7 days hire. Details S.A.E. please: C.C.L., 19 Darrell Way, Abingdon, OX14 1HQ.

SOFTWARE APPLICATIONS

HOME ACCOUNTS

Put your house in order! Probably home computing's best use! Comprehensive coverage of bank accounts. Credit cards, HP in-built accuracy check. Records all transactions. Projects cash-flow for any period ahead. Available for Complete Commodore Range. £8.45 or *free* details from:

Discus Software
Freepost
Beach Approach
Brixham TQ5 8BR
Tel: 08045 55532

GENERAL

BUY OR SELL your unwanted software, books, and hardware. We put you directly in contact with each other in your area. Details: Second Hand Computer Sales Agency. 38, Bowling Alley Walks, Dorchester, Dorset. Tel (0305) 64849 (24 hrs).

HARDWARE

STAR SG-10C PRINTER

ONLY £249 TO INCLUDE VAT & NEXT DAY DELIVERY BY SECURICOR

- ★ 100% compatible with your Commodore computer.
- ★ 120cps draft mode.
- ★ Tractor and friction feed standard.
- ★ Near Letter Quality mode included.
- ★ 192 graphic mode characters.
- ★ 192 business mode characters.
- ★ 90 italic characters.
- ★ 90 NLQ characters.
- ★ Normal 80 cpl. Condensed 136 cpl.
- ★ Enlarged 2X, 3X & 4X.
- ★ Supplied with interface/cable and mains plug READY TO GO.

SEIKOSHA GP100 VC PRINTER £99

STRONG COMPUTER SYSTEMS
BRYNA COTTAGE, PENIEL, CARMARTHEN, DYFED SA32 7DJ C64
TEL: (0267) 231246 FOR ASSISTANCE!!!

★ BACK-UP-SYSTEMS FOR CBM-64 & VIC-20 ★

The BACK-UP-BOARD Mk. 1 is the fastest, and easiest way to back up your tape programs. With this board you can connect a second CBM cassette deck to your computer. You can copy a program on the second deck while the original loads from the first. You can back up all fast loaders, data files, etc. BACK-UP-BOARD Mk. 1 £10.00 inc. P&P.

BACK-UP-BOARD Mk. 2. This new version of our famous BACK-UP-BOARD is now proving so popular it is outselling our original Mk. 1. This new board now works two ways. Switched to 'NORMAL' it operates as the Mk. 1. Switched to 'GENERATE', '64' owners can use our special transfer software to regenerate a copy of a program on the second deck as it loads from the first. This method can produce a better quality copy than the original. BACK-UP-BOARD Mk. 2 & Software £14.50 inc. P&P. Conversion service. Return your Mk. 1 Board, if purchased from us, to be modified to Mk. 2 for £4.50 including software & P&P.

AZIMUTH HEAD ALIGNMENT TAPE by INTERCEPTOR for '64'. Use this test tape with your '64' to adjust the heads of your decks to make a perfectly matched pair. Full instructions and screwdriver supplied. £8.50 inc. P&P.

ROM SWITCH/RESET BOARD. Fit this board in the expansion port of your '64' and you can reset ALL programs. Resets programs that a standard reset switch can't. A socket for a 2764 8K eprom is also fitted. Full instructions supplied. £8.95 inc. P&P.

Please send SAE for more details of the above products. Send your cheque/PO to:

TURBOTRONIC
46 Ripon St., Parkinson Lane, Halifax, West Yorkshire HX1 3UG.
Tel: (0422) 52020
(Europe add 50p outside Europe add £1.50)

TERMS

CLASSIFIED ADVERTISING TERMS & CONDITIONS

Our terms for new advertisers (semi-display and lineage) are strictly proforma payments until satisfactory reference can be taken up (excluding recognised advertising agencies). Cheques PO's should be crossed and made payable to:

ARGUS SPECIALIST PUBLICATIONS LTD.,

and send together with the advertisement to:
THE CLASSIFIED DEPT., L/H, NO: 1 GOLDEN SQUARE, LONDON W1R 3AB.

There are no reimburesments for cancellations. Advertisements arriving too late for a particular issue will be inserted in the following issue unless accompanied by instructions to the contrary.
All advertising sales are subject to Government regulations concerning VAT. Advertisers are responsible for complying with the various legal requirements in force eg: The Trade Description Act, Sex Discrimination Act & The Business Advertisements (Disclosure) Order 1977.

FULL TERMS & CONDITIONS OF ADVERTISING AVAILABLE ON REQUEST

FOR ADVERTISING DETAILS PHONE STUART TAYLOR ON 01-437-0626

FOR SALE

GADGET 64 £6.95
This 100% machine code CBM64 cassette utility offers: Append, Delete, Renumber (including Goto, Gosub, etc.) Satisfyingly simple to use. Send your Name, Address & PO or Cheque for £6.95 (P&P Free) to
J & J Gilmour Ltd.,
Dept YC486, 4 Copse Close, Liss, Hampshire GU33 7EW, England.
Tel: 0730-893430.

HARDWARE

Super Breaker Back-Up Board

For the Commodore 64, CBM 128 and VIC-20. When a second CBM type cassette unit is connected to the board it will make security back-up copies of cassette based software, including Turbo/Fast Loaders, Basic and/or Machine Code and Data Files. No additional software is needed. Very User Friendly, the back-up copy is made on the second cassette while the original is loading on the first. Price at £8.95 Plus P&P.

RESET CARTRIDGE for CBM 64. This cartridge will reset all, yes all, memory resident programs. Including all programs where standard reset switches fail. Price £7.95 + P&P.

P&P 55p UK (Europe add £1.00, elsewhere add £2.00). Send cheque/postal order, or a stamp for details to:
C. Kent, 26 Northcape Walk, Corby, Northants. NN18 9DQ.
For enquiries telephone (0536) 742622

ADVERTISING 01 437 0699

YOUR COMMODORE CLASSIFIED ADVERTISEMENT ORDER FORM

Advertise nationally in these columns for only 40p per word (min charge 15 words). Simply print your message in the coupon and send with your cheque or postal order made payable to **Argus Specialist Publications Ltd** to:

STUART TAYLOR
ASP LTD, 1 GOLDEN SQUARE, LONDON W1
Tel: 01-437 0699

Name ....

Address ....

....

....

....

....

Tel no (day) ....

....

ADVERTISERS PLEASE ADD 15% VAT

Please place my advert in **YOUR COMMODORE** for........issues.
Please indicate no. of insertions required.


THE COUNTDOWN HAS BEGUN
CRITICAL MASS
From DURELL
Commodore 64
Spectrum
R.R.P. £8.95
DURELL sales dept.,
Castle Lodge, Castle Green, Taunton TA1 4AB
Write now for details of Durell's
Great New Competition also
T shirts, Catalogue
and Posters

